

## THESIS / THÈSE

### MASTER EN SCIENCES INFORMATIQUES

#### Mise en place d'une méthodologie de développement au sein d'un service public fédéral: analyse de déroulement du projet d'implémentation

Pechon, Vincent

*Award date:*  
2006

[Link to publication](#)

#### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

#### Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

**FACULTES UNIVERSITAIRES NOTRE-DAMME DE LA PAIX, NAMUR**  
**INSTITUT D'INFORMATIQUE**  
**RUE GRANDGAGNAGE, 21, B-5000 NAMUR (BELGIUM)**

Mise en place d'une méthodologie  
de développement au sein d'un Service Public Fédéral

Analyse de déroulement du projet d'implémentation

Vincent PECHON

Mémoire présenté en vue de l'obtention du grade de

Licencié en informatique

Année académique 2005 - 2006



## **Résumé**

Le S.P.F. Finances (Service Public Fédéral, anciennement Ministère des finances) est actuellement confronté à une grande mutation organisationnelle et fonctionnelle suite aux réformes Copernic (modernisation des Administrations Publiques Fédérales).

La conséquence directe de la refonte des processus de travail et de la redéfinition des fonctions entraîne la nécessité d'adapter les programmes informatiques du S.P.F.

Le besoin de méthodologie de développement de programmes informatiques se fait cruellement ressentir.

Des groupes d'études ont été mis en place afin de déterminer les méthodologies les plus appropriées ainsi que les outils informatiques qui seront utilisés. Le fruit de leur travail a donné naissance à un projet nommé « Support au Développement » plus communément appelé « SupDev » dont la mission est d'implémenter une méthodologie de développement de projets informatiques.

L'objet de ce mémoire se concentre sur l'élaboration de cette méthodologie, son contenu et son implémentation.

Enfin, nous aborderons également la possibilité de réutiliser cette méthodologie au sein d'autres S.P.F. ou organismes d'Etat.

## **Mots clés**

RUP, UP, FUP, méthodologie de développement, implémentation, Copernic, CoperFin.

## **Abstract**

The "S.P.F. Finances (Federal Public Service, previously called "Ministry of Finances") is facing for the moment an important organisational and functional change following the Copernic reform(modernization of the Federal Public Administrations).

The direct consequence of the fusion of the working processes and the redefinition of functions leads to the necessity to adapt the IT programs of S.P.F.

A development methodology of IT programs is heavily needed.

Group studies have been implemented in order to determine the most appropriate methodologies as well as IT tools that will be used. The result of their work lead to a project called "Support to Development" more usually called "SupDev". The mission of "SupDev" is to implement a development methodology of IT projects.

The purpose of this study is to focus on the elaboration of this methodology, its content and its implementation.

Finally, we will also deal with the possibility to implement this methodology inside other S.P.F or Public Corporations.

## **Key words**

RUP, UP, FUP, development methodology, implementation, Copernic, CoperFin.

## Remerciements

Muriel, mon épouse est sans doute la personne qui doit être remerciée le plus chaleureusement. Sans elle, jamais, nous n'aurions pu aboutir au terme de ce mémoire et de ces longues années d'étude. Elle nous a encouragé et soutenu pour que finalement nous puissions mener à bien cette formation.

Merci à Emilie, Pauline et Quentin qui malgré leur jeune âge ont compris et accepté avec patience les longues absences de leur père.

Merci à Didier Leemans qui nous a donné l'occasion professionnelle d'explorer une problématique peu connue et qui nous a fait confiance pour mener à bien les missions confiées qui sont en partie à la base de ce mémoire.

Merci à Monsieur Habra, notre promoteur de mémoire pour son soutien et ses avis éclairés.

Merci aussi à nos compagnons de LIHD avec qui nous avons passé de longs moments studieux et pour qui la solidarité n'est pas un vain mot. Merci particulièrement à Daniel, Olivier et Benoît.

Nous voulons aussi remercier tous les professeurs, secrétaires et conseillères pédagogiques pour leur érudition, leur disponibilité et leur respect des étudiants.

Nous tenons à remercier également toutes les autres personnes qui nous ont aidé en relisant ce travail, en nous soutenant le moral par leur encouragement et leur compréhension, et particulièrement nos parents.



# Table des matières

<b>Glossaire</b>	<b>7</b>
<b>1. Introduction générale</b>	<b>13</b>
<b>2. Le contexte</b>	<b>15</b>
<b>2.1. La réforme Copernic</b>	<b>15</b>
2.1.1. Les enjeux stratégiques de la réforme	16
2.1.2. Les quatre axes de la réforme Copernic	16
<b>2.2. Le Service Public Fédéral Finances : présentation</b>	<b>21</b>
2.2.1. Introduction	21
2.2.2. La gestion des recettes et des dépenses de l'État	21
2.2.3. Les missions d'intérêt général	22
2.2.4. Des nouvelles méthodes de travail et une nouvelle culture administrative	22
2.2.5. Conclusion	22
<b>2.3. La réforme CoperFin : la rencontre des objectifs stratégiques de Copernic</b>	<b>23</b>
2.3.1. CoperFin : les BPR dans la pratique	23
2.3.2. Réforme organisationnelle	24
<b>2.4. Les conséquences de CoperFin</b>	<b>26</b>
2.4.1. Volet processus : la refonte des processus de travail	26
2.4.2. La modernisation du S.P.F. Finances : les aspects ICT	28
2.4.3. La situation de départ : état des lieux des logiciels de travail existants	33
2.4.4. L'adaptation des logiciels de travail suite à l'analyse de l'existant	34
<b>2.5. Les bases IT du S.P.F. Finances</b>	<b>35</b>
2.5.1. Contexte IT	35
2.5.2. Cadre spécifique	36
<b>3. Introduction d'une méthodologie de développement : le projet Support au Développement (SupDev)</b>	<b>37</b>
<b>3.1. Les enjeux</b>	<b>37</b>
<b>3.2. Les objectifs du projet</b>	<b>38</b>
3.2.1. Mettre en place un standard imposé à tous les intervenants	38
3.2.2. Maîtriser les coûts de développement	39
3.2.3. Mettre en place un cadre strict et porteur	39
3.2.4. Assurer la polyvalence	39
3.2.5. Maîtriser toutes les étapes et les productions	40
3.2.6. Assurer la prise en compte des aspects « métier » du développement	41
3.2.7. Assurer la maîtrise d'une suite logicielle permettant le soutien de la méthodologie par l'élaboration de modes opératoires précis	41
3.2.8. Assurer une évolution qualitative des développements	41
3.2.9. Assurer l'évolution qualitative de l'organisation	41
3.2.10. Mettre en place la centralisation des informations	41
<b>3.3. Le périmètre du projet SupDev</b>	<b>42</b>
3.3.1. Dans le périmètre du projet	42
3.3.2. Hors du périmètre du projet	42
<b>3.4. Les critères de réussite</b>	<b>43</b>
3.4.1. Mettre en place un standard imposé à tous les intervenants	43
3.4.2. Maîtriser les coûts de développement	43
3.4.3. Mettre en place un cadre strict et porteur	44
3.4.4. Assurer la polyvalence	44
3.4.5. Maîtriser toutes les étapes et les productions	45
3.4.6. Assurer la prise en compte des aspects « métier » du développement	45
3.4.7. Assurer la maîtrise d'une suite logicielle permettant le soutien de la méthodologie par l'élaboration de modes opératoires précis	45
3.4.8. Assurer une évolution qualitative des développements	46
3.4.9. Assurer l'évolution qualitative de l'organisation	46

<b>3.5. Risques</b>	<b>47</b>
3.5.1. Risques liés au contexte	47
3.5.2. Risques liées à l'ICT	47
<b>3.6. Facteurs critiques de succès</b>	<b>47</b>
<b>3.7. Hypothèses de départ</b>	<b>48</b>
<b>3.8. Les contraintes</b>	<b>49</b>
3.8.1. Organisation du service IT du S.P.F. Finances	49
3.8.2. Organisation du service d'analyse métier du S.P.F. Finances	49
3.8.3. Les différents types de projets et leur organisation	50
3.8.4. Mentalité du personnel d'un S.P.F.	54
3.8.5. Restructuration organisationnelle et fonctionnelle	55
3.8.6. Fonctionnement par appel à des sociétés externes	55
3.8.7. Intégration de l'approche métier et ICT	56
<b>3.9. Liens avec d'autres projets</b>	<b>56</b>
<b>3.10. Démarche</b>	<b>57</b>
3.10.1. Définition du processus de sélection	57
3.10.2. Evaluation des différentes méthodes candidates possibles	68
3.10.3. Choix de la méthode	69
3.10.4. Processus d'adaptation de la méthode pour répondre aux besoins du S.P.F. Finances.	69
<b>3.11. Rôles et responsabilités des parties prenantes</b>	<b>75</b>
<b>3.12. Budget</b>	<b>76</b>
<b>3.13. Structure d'accompagnement et d'encadrement</b>	<b>78</b>
3.13.1. Propriétaire du projet	78
3.13.2. Le comité de Direction	78
3.13.3. Le comité de Pilotage (Steering-committee)	78
<b>4. Etat d'avancement du développement de la méthodologie</b>	<b>80</b>
<b>4.1. Aspects Méthode</b>	<b>80</b>
4.1.1. Le cycle de vie	80
4.1.2. Les Disciplines	82
4.1.3. Les modes opératoires	84
4.1.4. Les concepts transversaux	85
<b>4.2. Avancement des Projets pilotes</b>	<b>87</b>
4.2.1. Critères de sélection	87
4.2.2. Projets retenus	88
4.2.3. Risques	89
4.2.4. Contraintes	90
4.2.5. Etat d'avancement des projets pilotes	90
<b>5. Les premiers enseignements retirés des projets pilotes en cours</b>	<b>92</b>
<b>5.1. Aspect Méthode</b>	<b>92</b>
5.1.1. Le modèle itératif	92
5.1.2. Le cycle de vie	93
5.1.3. Les livrables	94
5.1.4. Le Dictionnaire des données	94
<b>5.2. Aspect changement</b>	<b>95</b>
5.2.1. Besoin de formation	95
5.2.2. Transfert de connaissances et d'expériences	95
<b>5.3. Aspect humain</b>	<b>96</b>
5.3.1. Absence de structures organisationnelles soutenant les processus de développement de FUP	96
5.3.2. Suivi plus proche des ressources	96
5.3.3. Mieux exprimer les enjeux stratégiques de l'adoption d'une méthodologie	96
<b>5.4. Aspect projet</b>	<b>97</b>
5.4.1. Besoin d'une méthodologie de gestion de projet	97
5.4.2. Besoins d'un référentiel modélisé des standards de développement IT	97



<b>6. Recommandations à l'implémentation de la méthode au S.P.F Finances</b>	<b>99</b>
6.1. Aspect organisation : mettre en application effective la réforme CoperFin.	99
6.2. Aspect changement	100
6.2.1. Eviter le Big Bang.	100
6.2.2. Veiller à la cohérence du contrôle, de la gestion et du développement d'un projet	101
6.3. Aspect humain	103
6.3.1. Renforcer la gestion des ressources humaines	103
6.3.2. Recruter des ressources avec des profils professionnels définis	103
6.3.3. Veiller à la formation continue des différents acteurs	104
6.4. Aspect projet	104
6.4.1. Assurer le partage des informations transversales entre les projets	104
6.4.2. Synchroniser les plannings	105
<b>7. Evaluation du projet SupDev en fonction des critères de réussite définis précédemment (section 3.4)</b>	<b>106</b>
7.1. Mettre en place un standard imposé à tous les intervenants	106
7.2. Maîtriser les coûts de développement	106
7.3. Mettre en place un cadre strict et porteur	106
7.4. Assurer la polyvalence	107
7.5. Maîtriser toutes les étapes et les productions	107
7.6. Assurer la prise en compte des aspects « métier » du développement	107
7.7. Assurer la maîtrise d'une suite logicielle permettant le soutien de la méthodologie par l'élaboration de modes opératoires précis	107
7.8. Assurer une évolution qualitative des développements	108
7.9. Assurer l'évolution qualitative de l'organisation	108
<b>8. Evolution de la méthodologie</b>	<b>109</b>
8.1. Adaptation aux spécificités des projets	109
8.2. Adaptation à la maturité de l'organisation	110
8.3. Adaptation à l'implémentation de la gestion de projet	111
<b>9. Réutilisabilité de la méthodologie FUP dans le cadre d'autres organisations</b>	<b>112</b>
<b>10. Conclusion</b>	<b>114</b>
<b>11. Bibliographie</b>	<b>115</b>
<b>12. Annexes</b>	<b>117</b>
12.1. Rapport Evaluation Support Au Développement.	117
12.2. Les outils de développement et de modélisation, soutiens à la méthodologie	127
12.3. Les conventions de nommage	138
12.4. Exemples de la méthodologie FUP	153
12.4.1. Discipline modélisation métier	153
12.4.2. Discipline modélisation métier – Mode opératoire – Présentation générale	173
12.4.3. Discipline modélisation métier – Mode opératoire détaillé – processus métier	184

## Table des illustrations

Figure 2-1 : Le logo officiel de Copernic	15
Figure 2-2 : Les quatre axes de la réforme Copernic	16
Figure 2-3 : La transformation des ministères en Services Publics Fédéraux (S.P.F.)	17
Figure 2-4 : Le système des mandats	18
Figure 2-5 : Le business process re-engineering	20
Figure 2-6: La juste perception de l'impôt (extrait du rapport annuel 2003)	21
Figure 2-7 : Logo officiel CoperFin	23
Figure 2-8 : Elaboration conceptuelle d'un nouvel organigramme pour le SPF Finances	24
Figure 2-9 : Les nouveaux processus	27
Figure 2-10 : Les fondements ICT et «métier» du S.P.F. Finances	35
Figure 3-1 : Les différents standards au S.P.F. Finances	39
Figure 3-2 : La couverture méthodologique	40
Figure 3-3: Organisation du service ICT.	49
Figure 3-4: Découpage « préétude » / « projet »	54
Figure 3-5 : Approche classique et orientée objet	57
Figure 3-6 : Les principaux livrables selon l'approche	58
Figure 3-7 : L'approche classique – données – processus/programme	59
Figure 3-8 : De l'analyse structurée à la programmation structurée	60
Figure 3-9 : La dynamique de l'objet	60
Figure 3-10 : Adéquation des méthodes agiles en fonction de la taille du projet	66
Figure 3-11 : Disposition légales en matière de propriété	78
Figure 4-1 : Le cycle de vie version S.P.F. Finances	80
Figure 4-2 : Le cycle de vie de FUP inspiré de RUP	80
Figure 4-3 : Les principaux jalons du cycle de vie	80
Figure 5-1 : Le cycle de vie d'un projet externalisé	93
Figure 6-1 : Organisation des PMO's	101
Figure 6-2 : Contrôle, gestion et développement d'un projet	102
Figure 12-1 : La suite logicielle retenue	127



## Glossaire

Termes	Explication
<b>A</b>	
Administration générale	Les Administrations générales se situent au niveau hiérarchique N-1 du nouvel organigramme du S.P.F. Finances, et rapportent au Président du S.P.F. (niveau N). Il existe 3 Administrations générales: "Impôts et Recouvrement", "Documentation patrimoniale", et "Trésorerie".
AFER	Administration de la fiscalité des entreprises et des revenus.
Analyse "As Is"	"As Is" signifie "existant". Analyse de la situation comme elle se présente aujourd'hui.
Analyse "To Be"	"To Be" signifie "futur". Analyse de la situation comme elle devrait se présenter demain.
Appel d'offre	Publication d'un cahier des charges auprès de sociétés extérieures afin de recevoir leur proposition de réalisation du projet moyennant la passation d'un contrat.
Assessment	"Assessment" signifie "évaluation, estimation". Méthode qui permet d'évaluer les candidats à une fonction sur base de leurs compétences, en utilisant des critères bien définis. L'Assessment constitue un outil de diagnostic permettant de mettre en lumière les forces et les faiblesses des personnes évaluées, et se base souvent sur des tests de simulation et des exercices liés à la fonction future du candidat. Il en va de même pour l'analyse d'une organisation ou d'un service au niveau des processus de travail et de l'organisation de ces derniers.
<b>B</b>	
Back office	Partie de l'organisation qui fournit des services sur le plan du contenu (en soutien du Front Office - voir ci-dessous). Cette unité travaille uniquement pour des "clients" internes et entre peu ou pas en contact avec les "clients" externes.
Bloc d'activités	Groupe d'activités qui ont un lien logique entre elles. Plusieurs blocs d'activités forment ensemble un processus.
BPR (Business Process Reengineering)	Un projet "Business Process Reengineering" détermine l'ensemble des changements nécessaires au niveau du personnel, des moyens (informatiques) et des méthodes de travail pour créer un meilleur environnement de travail et pour atteindre de meilleurs résultats. Le BPR regroupe l'ensemble des programmes et des réseaux chargés de dessiner l'image future du fonctionnement du Ministère des Finances. Ils vont proposer des améliorations concrètes pour les procédures et les méthodes de travail des entités fiscales des Finances.



Business Case	"Business case" signifie "Etude de cas". Modèle qualitatif, reprenant les bénéfices et les coûts, utilisé pour décider de l'approbation d'un investissement ou d'un changement à réaliser. Les Business Cases sont souvent utilisés lors d'un trajet de changement.
C	
Call center	"Call center" signifie "Centre téléphonique". Canal d'interaction centralisé (pas nécessairement à un endroit) destiné à recevoir les appels téléphoniques des citoyens et à y répondre. Un contact physique entre le citoyen et l'organisation n'est pas possible par ce biais.
Canal d'interaction	Canal ou chemin par lequel a lieu l'interaction ou le contact entre le citoyen et le Ministère - c'est via ce canal que les services et l'information sont fournis au client ou que le client peut demander de l'information.
CCFF	Acronyme de Centre de Communication de la Fiscalité fédérale ;futur point de contact unique entre le S.P.F. Finances et ses clients
Change Assessment	"Change Assessment" signifie "Evaluation du changement". Examen de l'adoption et de l'appropriation par l'organisation de ses nouvelles capacités (ici, procédures, structure, culture, méthodes de travail, etc). Cette évaluation a lieu après la mise en œuvre du changement.
Change Management	"Change Management" signifie "Gestion du changement". L'accompagnement d'un changement afin de parvenir à une meilleure acceptation de ce changement par le public-cible.
Comité de pilotage	Le comité de pilotage dirige et suit les travaux de Coperfin et est chargé de valider tous les résultats. Il détermine si le chemin emprunté par chaque groupe de travail est correct. Si nécessaire, le comité prend les décisions pour régler les problèmes. Dans le cadre de Coperfin, le comité de pilotage se compose du Secrétaire général, de différents directeurs généraux et de représentants du Ministre, du Commissaire du Gouvernement, du Ministre de la fonction publique et de FEDICT.
Comité directeur	Le Comité directeur est l'organe hiérarchique le plus élevé dans la structure fiscale du S.P.F. Finances. Le Comité prend toutes les décisions finales pour CoperFin. Il est présidé par le Secrétaire général et se compose des Directeurs généraux des administrations fiscales, du Directeur général des Services Généraux et des représentants du Cabinet.
Compétence	Savoir-faire et connaissances nécessaires à l'exécution d'une fonction définie.
Compréhension du client	A l'aide de méthodes d'analyse, les données brutes des clients sont transformées en données qui procurent une compréhension et une connaissance de la situation, des besoins du client. Sur base de ces données, les clients sont répartis en groupes-cibles.
Consolider	Assembler différents éléments dans le but de parvenir à un ensemble cohérent.



Contribuable	Personne physique ou morale qui paye des contributions, des impôts directs.
CoperFin	CoperFin est le nom donné à la modernisation du ministère des Finances, mise en œuvre par les fonctionnaires des Finances. CoperFin ne peut être vu séparément des autres initiatives déjà en cours: nouvel organigramme, management de la performance, contrôle et audit internes. CoperFin renvoie en premier lieu aux 16 programmes BPR qui dessinent la nouvelle image future du ministère des Finances pour les Administrations générales Impôts et Recouvrement et Documentation patrimoniale. Ces programmes vont de pair avec la modernisation au niveau de l'infrastructure, de l'ICT et du personnel.
Copernic	Projet de réforme de l'Administration fédérale belge
Core Business	"Core Business" signifie "Activité, occupation principale".
CRM	"Citizen Relationship Management" signifie "Gestion des relations avec le citoyen". Le Ministère désire améliorer la qualité de ses relations avec les citoyens/contribuables et être capable de mieux répondre à leurs attentes.
<b>D</b>	
Deliverable	Information documentée et connaissance produite comme résultat d'un processus méthodologique qui doit généralement être présentée dans un délai déterminé à l'avance.
Description fonctionnelle d'une application	Les capacités ou comportements d'un programme, partie d'un programme, ou système, vu comme la somme de ces caractéristiques. En bref, c'est ce que le système devrait être capable de faire, les fonctions qu'il devrait remplir.
Direction générale	Les Directions générales se situent au niveau hiérarchique N-2 du nouvel organigramme du S.P.F. Finances, et rapportent aux Administrations générales (niveau N-1). Chaque Administration générale se divise en plusieurs Directions générales. Par exemple, l'Administration générale "Documentation patrimoniale" est composée de 4 Directions générales: "Services patrimoniaux", "Mesures et Evaluations", "Collecte et échange d'informations" et "Sécurité juridique".
Dossier fiscal unique	Rassemblement dans un dossier, accessible de manière automatisée, de tous les éléments disponibles (aussi bien fiscaux que non-fiscaux) qui se rapportent à un contribuable.
<b>E</b>	
Expertise	Compétence accrue dans un domaine défini.
<b>F</b>	
FAQ	"Frequently Asked Questions" signifie "Questions fréquemment posées". Liste des questions les plus fréquemment posées avec leurs réponses.
FEDICT	Acronyme de FEDeral ICT ; S.P.F. horizontal chargé de la coordination générale informatique pour tous les SPF.
Fonction	Groupe de tâches qui peuvent être exécutées par une personne.



Front office	Partie de l'organisation qui interagit directement avec le citoyen. Elle représente la première ligne de contact quand le citoyen désire entrer en contact avec l'administration. Cette partie reçoit le soutien du back office.
FUP	Nom de la méthodologie de développement implémentée au S.P.F. Finances (Financial Unified Process)
G	
Groupe de travail	Le groupe de travail est composé de fonctionnaires issus tant des services centraux qu'extérieurs. Les membres ont été sélectionnés sur base de leur expertise fonctionnelle ou sur base de leur connaissance approfondie de certains aspects des processus centraux, ainsi que pour leur sensibilité du terrain. Les groupes de travail dessinent le fonctionnement futur des différentes administrations.
H	
I	
ICT	"Information and Communication Technologies" signifie "Technologies de l'Information et de la Communication" (TIC). L'ICT regroupe le matériel, les systèmes et les applications informatiques.
Implémentation	L'exécution (ou encore la mise en œuvre) d'un plan, d'un nouveau système dans une organisation.
Indicateur de risque	Facteur ou type d'information qui expriment le(s) risque(s) et qui, pris isolément ou mis en corrélation avec d'autres indicateurs, après analyse et évaluation, permet de déterminer le niveau de risque.
Infrastructure	Immobilier, c'est à dire bâtiment, avec toutes ses caractéristiques (principalement espaces de bureau, salles de réunion, parking, endroit de stockage, équipement de communication et réseau IT,...). Dans le cadre du groupe de travail "Infrastructure", l'analyse ne porte pas sur les éléments suivants: les meubles, les équipements de bureau, le matériel informatique (software et hardware),...
Input	Dans le cadre d'un processus, produit ou élément d'information nécessaire à l'accomplissement d'une tâche.
Interface	Lien avec un autre processus au sein-même du programme ou avec un processus d'un autre programme.
Interopérabilité	Capacité de plusieurs systèmes à coopérer harmonieusement.
J	
K	
L	
Livrables	Synonyme de « délivrables » (voir délivrables)
M	
Manager de programme	Le manager de programme endosse la responsabilité finale du planning et de la direction des travaux au sein du programme.
Marché public	Marché proposé à toute société externe au S.P.F. Finances pour la réalisation d'un projet sur base d'un cahier des charges. Les sociétés intéressées soumissionnent le marché et après analyse des offres, le marché est attribué à la société répondant au mieux aux desiderata du cahier des charges.



<b>N</b>	
Niveau N	Niveau hiérarchique le plus élevé du S.P.F. Finances (Président du S.P.F.).
Niveau N-1	Niveau hiérarchique du S.P.F. Finances qui rapporte au niveau N (Administrateur général).
Niveau N-2	Niveau hiérarchique du S.P.F. Finances qui rapporte au niveau N-1 (Directeur général).
Niveau N-3	Niveau hiérarchique du S.P.F. Finances qui rapporte au niveau N-2 (Directeur de Centre).
<b>O</b>	
Output	Dans le cadre d'un processus, produit ou élément d'information qui est le résultat d'une tâche.
<b>P</b>	
PMO	Acronyme de Program Management Office ; structure ou cellule, composée de représentants d'ICT et des unités opérationnelles clientes, chargée de suivre et coordonner les différents projets et de faire rapport au haut management.
Prestation de services	Le fait de rendre un service (produit, information, ...) à une contrepartie (client, citoyen, contribuable).
Principes stratégiques	Principes qui serviront de base, de fil rouge, au fonctionnement des futures Directions générales du S.P.F. Finances.
Proactif	Adopter un certain comportement, avant qu'un événement se produise, en anticipation de celui-ci
Processus	Ensemble d'activités répétitives liées entre elles qui apportent de la valeur pour l'organisation et ses clients. Un processus se compose de plusieurs blocs d'activités
Progiciel	Elément de logiciel "prêt-à-porter" (par opposition aux applications "sur mesures" développées spécifiquement pour un client donné).
<b>Q</b>	
<b>R</b>	
Réactif	Adopter un certain comportement, après qu'un événement se soit produit, en réaction à celui-ci.
Reengineering	Voir Business Process Reengineering
Release	Etape intermédiaire, version d'un logiciel.
Résultat	Ce sur quoi un processus débouche. Ceci signifie que l'objectif du processus est atteint. Un processus peut avoir plusieurs résultats.
Risque	Possibilité qu'un dispositif légal ou réglementaire ne soit pas appliqué, ou soit appliqué de manière incomplète ou inexacte, tant par l'Administration que par un opérateur fiscal ou un groupe d'opérateurs fiscaux, que ce soit en faveur ou en défaveur soit du trésor, soit de l'opérateur ou du groupe d'opérateurs. En matière de recouvrement, possibilité qu'une créance fiscale ne soit pas recouvrée pour ce qui est recouvrable, en fonction de la solvabilité telle qu'elle apparaît du dossier unique recouvrement.



Roll-Out	"Roll-out" signifie au sens large "déploiement".Le roll-out est une approche globale qui consiste non seulement à livrer aux utilisateurs des nouveaux outils informatiques modernes et performants, mais également d'installer ces ordinateurs et de les munir des logiciels nécessaires à la personne pour effectuer son travail quotidien. Ce roll-out peut également s'appliquer aux imprimantes et autres périphériques (scanner,...).
S	
Scope	"Scope" signifie "Etendue, portée», délimitation d'un projet au niveau de son contenu..
Services	Produit non physique qui est délivré à un client interne ou externe (ex: information).
SLA	Acronyme de Service Level Agreement ; accord entre l'ICT et les entités opérationnelles clientes, sur un niveau de service à garantir(par exemple : taux de disponibilité, taux de panne, temps de réparation, etc.).
S.P.F.	Service Public Fédéral. Les ministères fédéraux vont tous progressivement passer vers la nouvelle structure de Service Public Fédéral. Pour le S.P.F. Finances, l'organigramme a déjà été dessiné jusqu'au niveau N-3.
Sponsor	Le sponsor soutient le trajet de changement et le réseau formé par les agents de changement. Il a l'autorité hiérarchique pour prendre des décisions pour forcer l'implémentation du changement. Chaque programme BPR bénéficie du soutien d'un sponsor. Le sponsor est responsable de l'application des décisions prises par le comité de pilotage, dont il fait partie. Il encouragera les changements et les actions à prendre au niveau du ou des programmes dont il est responsable.
Stakeholder	Personne ou groupe de personnes internes ou externes à un processus ou à un changement ayant un intérêt dans le déroulement de ce processus ou ce changement.
SupDev	Acronyme utilisé pour « Support au Développement », nom donné au projet d'implémentation d'une méthodologie de développement au S.P.F. Finances ainsi que d'une suite logicielle de modélisation des développements.
T	
Template	Document (électronique) standard dans lequel les styles, les polices et la mise en page sont prédéfinis.



# 1. Introduction générale

Dans une organisation aussi importante que celle des Services Publics Fédéraux Belges, regroupant quelques soixante mille fonctionnaires, les processus de travail sont régis par une panoplie de règlements internes traduits sous la forme de circulaire et d'instructions. Des programmes informatiques ont été développés afin d'aider les fonctionnaires dans leurs tâches et leur permettre d'accéder à des systèmes d'informations de plus en plus performants. Cependant, le citoyen, au cœur du système, n'arrivait plus à comprendre la complexité des démarches administratives.

Au début de l'année 2000, le Gouvernement Belge a entamé une profonde réforme de ses institutions étatiques afin d'y simplifier les méthodes de travail et d'y instituer une nouvelle culture de gestion. Le citoyen devait être au cœur des préoccupations. La réforme Copernic était lancée.

Au sein du Service Public Fédéral Finances (S.P.F. Finances), le plus important Ministère Belge puisqu'il compte à lui seul trente mille agents, les principes de Copernic ont donné naissance à plus grande réforme organisationnelle et fonctionnelle jamais conduite. Cette réforme a été traduite sous le vocable de réforme CoperFin (Copernic pour les Finances).

CoperFin a engendré une refonte totale des processus de travail et la redéfinition des différentes fonctions au sein de l'institution. L'organisation est aussi totalement changée, tant au niveau de son mode de fonctionnement qu'au niveau de son déploiement géographique.

Travaillant au sein du S.P.F. Finances ou Cabinet de l'Administrateur Général des Impôts et du Recouvrement, nous avons été pleinement concerné par cette réforme puisqu'il appartient à ce service central de la rendre effective.

Afin de soutenir les nouveaux processus de travail, un vaste programme d'investissements informatiques a été mis en place, concernant aussi bien le développement applicatif que l'acquisition de matériel performant permettant de répondre au mieux aux défis lancés par CoperFin.

Le nombre de projets ainsi initiés est fort impressionnant tant par leur ampleur que par leur nombre. Cependant l'organisation et les méthodes de travail des services d'analyses et d'ICT n'étaient pas adaptées à ces vastes chantiers. Il fut alors décidé par le management du S.P.F. Finances de mettre en chantier des projets de standardisation des méthodes de travail de ces services par l'implémentation entre autres d'une méthodologie de développement de projet. Le projet « Support au Développement » (en abrégé SupDev) a donc été initié.

Dans le cadre de ce projet nous avons été chargé de mener à bien l'analyse des méthodologies candidates existantes, d'en choisir une, de l'adapter à l'organisation future et de l'implémenter.

Dans le cadre de ce mémoire nous exposerons en détail les changements induits par les réformes Copernic et CoperFin dans l'organisation du S.P.F. Finances.

Ensuite nous analyserons le déroulement des différentes étapes qui ont jalonné le projet SupDev après en avoir défini les enjeux, les objectifs et les critères de réussite. Nous en développerons l'état d'avancement de la méthodologie et son implémentation dans deux projets pilotes.

Puis, nous analyserons les premiers enseignements retirés de ces projets pilotes et nous formulerons une série de recommandations à l'implémentation de la méthodologie. Dans la suite de cette analyse nous proposerons une évaluation du projet dans son ensemble et nous décrirons les évolutions adaptatives et correctives envisagées pour l'implémentation finale de la méthode à toute l'organisation.

Enfin, nous envisagerons les possibilités de réutiliser les concepts méthodologiques dans le cadre d'autres organisations.



## 2. Le contexte

### 2.1. La réforme Copernic

Copernic désigne le projet de réforme de l'Administration fédérale belge. Il est destiné à améliorer les services que celle-ci fournit aux citoyens, ainsi que les conditions de travail et les perspectives de carrière des fonctionnaires.

Le principe de cette réforme est que le citoyen occupe une place centrale au sein de l'État. L'Administration doit lui offrir les services de qualité auxquels il a droit. Elle doit donc évoluer avec son époque et s'adapter aux réalités de la société contemporaine.



Figure 2-1 : Le logo officiel de Copernic

#### Pourquoi Copernic ?

Il est de tradition de baptiser les accords du nom de l'endroit où ils ont été signés ou de celui du saint fêté le jour de leur signature.

Le plan de réforme de l'Administration belge a été surnommé « plan Copernic » parce que le jour de sa présentation à la presse, le 16 février 2000, coïncidait presque avec la date anniversaire de la naissance du pionnier de l'astronomie moderne, le Polonais Nicolas Copernic, né le 19 février 1473.

À son époque, celui-ci avait proclamé l'idée révolutionnaire que la Terre tournait autour du Soleil et non le contraire.

Son nom a semblé tout indiqué pour caractériser la réforme de l'Administration fédérale, qui, en partant du principe que l'État doit être au service du citoyen, et non l'inverse, vise à induire un changement radical des mentalités.

### 2.1.1. Les enjeux stratégiques de la réforme

Le monde moderne évolue rapidement. Ces vingt dernières années, la croissance économique, les avancées technologiques et les progrès de l'enseignement ont amélioré la prospérité et ont profondément modifié notre façon de vivre.

Aujourd'hui, la population veut avoir son mot à dire et réclame des services d'une qualité optimale. Ceux qui fournissent ces services doivent s'adapter à cette nouvelle réalité.

L'Administration qui est le plus gros fournisseur de services se devait elle aussi d'évoluer.

En Belgique, les réformes constitutionnelles qui se succèdent depuis trente ans ont modifié le paysage institutionnel du pays et l'ont transformé en un État fédéral. Cependant, sa fonction publique était restée globalement inchangée et pâtissait d'un manque d'investissements. Sa structure, comme celle de beaucoup d'administrations, n'était pas ouverte au changement.

Le plan Copernic trace la voie à suivre pour mettre cette réforme en œuvre. Il s'agit d'un programme vaste et ambitieux de modernisation de l'Administration fédérale qui, avec plus de 60.000 agents, est aussi le plus gros employeur du pays.

### 2.1.2. Les quatre axes de la réforme Copernic

Pour atteindre le double objectif du meilleur service au citoyen et du meilleur cadre de travail des fonctionnaires, la réforme de l'Administration fédérale s'articule autour de quatre axes.

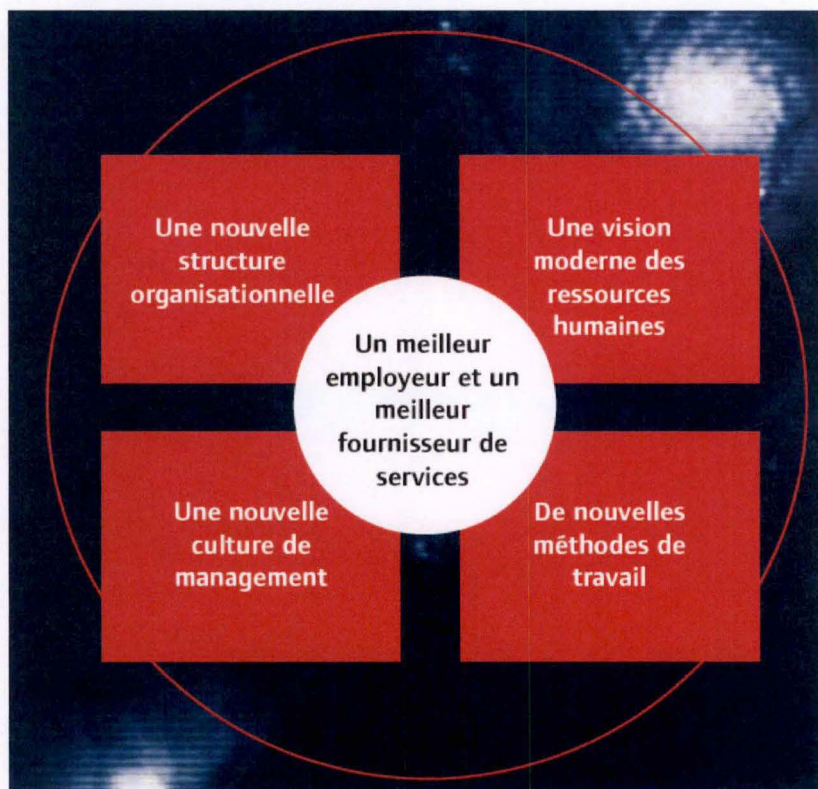


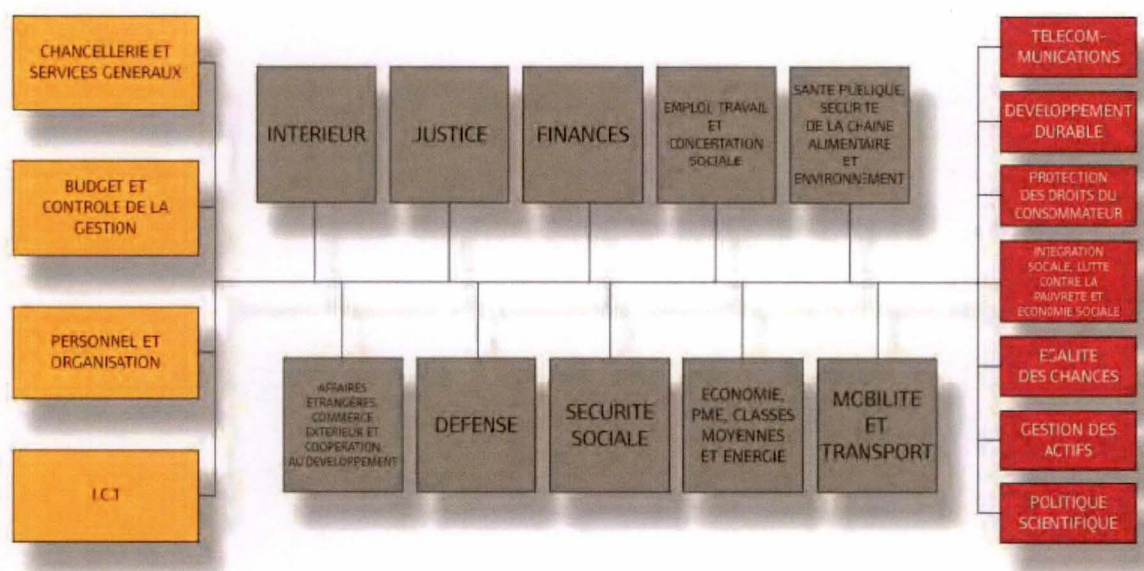
Figure 2-2 : Les quatre axes de la réforme Copernic



- Une nouvelle structure organisationnelle

### La transformation des ministères en services publics fédéraux

Les ministères fédéraux ont été soit scindés, soit fusionnés pour former les services publics fédéraux (S.P.F.). Chacun de ces services s'occupe d'un ensemble homogène de matières et dépend d'un seul ministre.



**Figure 2-3 : La transformation des ministères en Services Publics Fédéraux (S.P.F.)**

Il existe dix S.P.F. « verticaux », chargés chacun d'une politique spécifique, comme les Finances, l'Intérieur, la Mobilité et les Transports, l'Emploi et la Concertation sociale. Ces matières ont été restructurées de façon à englober un ensemble plus cohérent de compétences. Désormais, les citoyens en quête d'un service précis ne devront plus frapper qu'à une seule porte.

À ces S.P.F. verticaux s'ajoutent quatre S.P.F. « horizontaux ». L'un d'eux, le S.P.F. Chancellerie et Services généraux, est rattaché aux services du Premier ministre et joue un rôle de coordination.

Les trois autres s'occupent de matières communes à tous les S.P.F. : Budget et Contrôle de la gestion, Personnel et Organisation, Technologies de l'information et de la communication.

Ils déterminent la politique dans leurs domaines respectifs et assistent les autres S.P.F. pour ce qui relève de leurs compétences.

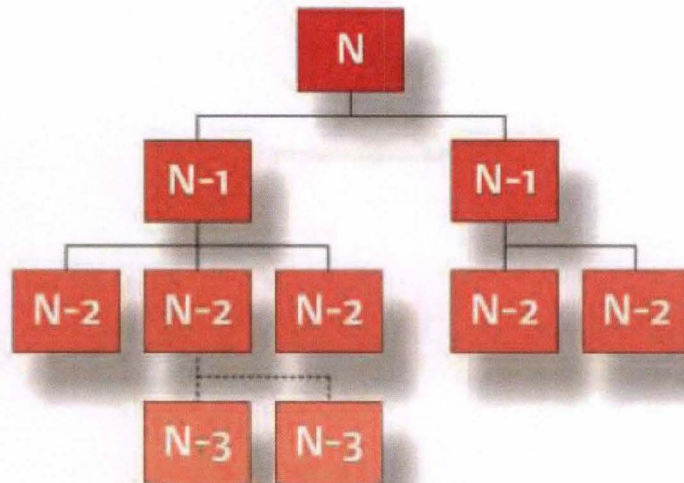
En outre, plusieurs services publics de programmation (SPP) ont été créés. Ils s'occupent de dossiers spécifiques ponctuels liés à des enjeux de société qui nécessitent une coordination entre plusieurs S.P.F., comme la politique d'égalité des chances ou le développement durable.

Chaque S.P.F. vertical possède ses propres services d'encadrement, qui mettent en application dans leur service les trois matières suivantes : Budget et Contrôle de la gestion, Personnel et Organisation et Technologies de l'information et de la communication.

- Une nouvelle culture de management

### Les principaux changements

- ↳ Le recrutement de top managers selon un système de mandats de 6 ans



Les mandats s'appliquent aux trois premiers niveaux du management (N, N-1, N-2), sauf au sein du SPF Finances, qui est de loin le plus gros service, où ils s'appliquent aussi au niveau N-3.

**Figure 2-4 : Le système des mandats**

- ↳ La mise en place de mécanismes de contrôle efficaces qui garantissent aux managers une liberté d'action dans un cadre précis.
- ↳ Plus de liberté et plus de responsabilité pour les S.P.F. dans l'utilisation de leur budget grâce à des procédures de contrôle budgétaire renouvelées et assouplies.
- ↳ Une meilleure information des fonctionnaires quant à leur rôle et aux changements en cours.
- ↳ Un rôle majeur pour les managers dans la nouvelle procédure d'évaluation du personnel en fonction de leurs compétences réelles.

### Pourquoi ces changements ?

- ↳ La procédure de sélection des actuels fonctionnaires dirigeants évaluait leurs compétences techniques et pas suffisamment leurs aptitudes à la gestion.
- ↳ La structure hiérarchique était trop dense et trop rigide.
- ↳ Il était nécessaire d'instaurer une communication dans les deux sens entre les fonctionnaires dirigeants et le personnel.



- Une nouvelle vision des ressources humaines

#### Les principaux changements

- ↗ Une plus grande attention au développement des compétences et aux perspectives de carrière du personnel.
- ↗ La revalorisation des compétences, l'amélioration de la satisfaction professionnelle.
- ↗ Faire de l'Administration publique un meilleur employeur.

#### Pourquoi ces changements ?

- ↗ L'affectation des agents doit tenir compte de leurs compétences pratiques et pas uniquement de leurs connaissances théoriques.
- ↗ Les carrières et les salaires des agents doivent évoluer parallèlement au développement de leurs compétences, plutôt que sur la base d'examens théoriques ou de leur ancienneté.
- ↗ L'ancien ministère de la Fonction publique s'occupait essentiellement de l'administration du personnel. Son organisation ne lui permettait pas de mener une véritable politique des ressources humaines.

- Une nouvelle méthode de travail

### Les principaux changements

LE « BUSINESS PROCESS RE-ENGINEERING » EST AU COEUR DE COPERNIC.

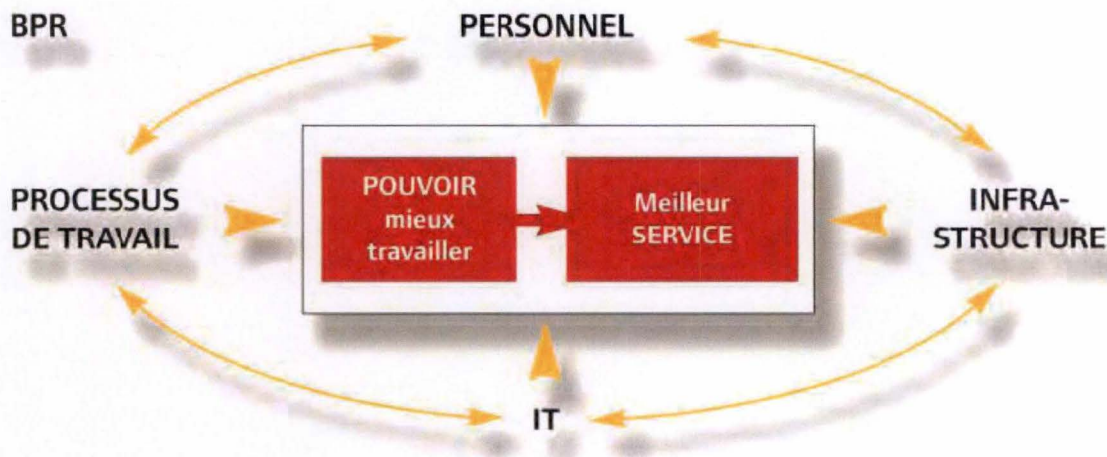


Figure 2-5 : Le business process re-engineering

Le « **business process re-engineering** » (BPR) s'applique à tous les aspects de la modernisation. Il consiste à examiner la manière dont les moyens actuels ou prévus (personnel, infrastructures, systèmes et processus) peuvent être utilisés pour améliorer les services et créer un cadre de travail moderne et professionnel. Les programmes de BPR, qui succèdent aux grandes modifications organisationnelles, permettent de mettre en œuvre les changements fondamentaux qui sont au cœur de la réforme Copernic

Des investissements seront consentis en faveur :

- ↗ du personnel : formation et développement de carrière
- ↗ des infrastructures : bâtiments, bureaux, matériel et équipement
- ↗ des systèmes (IT) : technologies de l'information, administration électronique, systèmes d'assistance
- ↗ des processus : des processus de meilleure qualité, plus souples et plus rentables pour de meilleurs services.

### Pourquoi ces changements ?

- ↗ Une longue période de sous-investissement dans les infrastructures a occasionné un immobilisme et une routine qui ne sont plus en phase avec la société actuelle.
- ↗ La nécessité d'investir dans les compétences des fonctionnaires, de manière à leur permettre de s'adapter à des méthodes de travail neuves et plus efficaces.



## **2.2. Le Service Public Fédéral Finances : présentation**

### **2.2.1. Introduction**

La gestion d'un Etat moderne passe par la satisfaction de besoins collectifs.

Le Service Public Fédéral Finances a pour mission principale d'assurer la collecte et la gestion d'une part importante des moyens nécessaires à la satisfaction de ces besoins.

Le principe du consentement à l'impôt institué par le Parlement constitue l'un des fondements de la démocratie. En acceptant de payer des impôts, les citoyens expriment leur volonté de vivre dans une société solidaire.

Le Service Public Fédéral Finances assure une juste et exacte perception de l'impôt, ce qui signifie que chaque contribuable supporte l'impôt légalement dû par lui.

**ASSURER LA JUSTE ET EXACTE  
PERCEPTION DE L'IMPÔT :  
NI PLUS, NI MOINS**

Contrairement à ce que d'aucuns pensent parfois, l'objectif du Service Public Fédéral Finances n'est pas d'améliorer le rendement des impôts en établissant des suppléments aussi élevés que possible.

Notre mission est de veiller à assurer une juste et exacte perception de l'impôt, tel que celui-ci a été consenti par le Parlement.

Le contrôle fiscal vise donc à ce que chacun supporte l'impôt légalement dû : ni plus, ni moins.

En d'autres termes, un contrôleur qui attire l'attention sur le fait qu'un contribuable risque de payer davantage que l'impôt dû (par erreur, inadvertance ou méconnaissance de la loi) accomplit un excellent travail.

Même si de tels cas demeureront moins fréquents que ceux dans lesquels un supplément d'impôt sera établi, le fait que chaque hypothèse soit désormais mise en exergue a le mérite de contribuer à l'indispensable évolution des mentalités et de rappeler le fondement constitutionnel de notre mission : le principe du consentement à l'impôt.

**Figure 2-6: La juste perception de l'impôt (extrait du rapport annuel 2003)**

### **2.2.2. La gestion des recettes et des dépenses de l'État**

L'Administration de la Trésorerie assure l'équilibre entre les recettes et les dépenses pour le compte de l'Etat fédéral. Dans ce cadre, elle comble les déficits en trouvant les moyens financiers, par voie d'emprunt et/ou de placement des surplus.

Dans le prolongement de cette mission, elle doit effectuer avec exactitude le paiement des dépenses générales, notamment les rémunérations du personnel de l'Etat et les remboursements d'impôts.

Les missions de la Trésorerie couvrent également un vaste volet européen et international en matière économique, financière et monétaire.

### **2.2.3. Les missions d'intérêt général**

Le Service Public Fédéral Finances assume aussi d'autres missions importantes d'intérêt général.

Assurant la tenue de la Documentation Patrimoniale, il contribue à garantir la sécurité juridique, notamment dans le cadre de la circulation juridique des biens.

À l'occasion du contrôle des marchandises effectué par la Douane, il participe aussi à la protection de la santé publique, de l'environnement et de la sécurité des personnes et des biens, notamment dans le cadre de la lutte contre les trafics illicites et le terrorisme.

### **2.2.4. Des nouvelles méthodes de travail et une nouvelle culture administrative**

La mission du Service Public Fédéral Finances s'articule sur le délicat équilibre entre les droits et les devoirs que la loi reconnaît ou impose au citoyen. Celui-ci est au cœur de sa mission.

Dans cette optique le Service Public Fédéral Finances se doit de relever les défis d'une administration moderne, en phase avec l'évolution de la société, de l'économie et des technologies de l'information et de la communication.

Dans le souci constant du service public, le Service Public Fédéral Finances développe et garantit une approche professionnelle et rigoureuse de ses métiers de base par :

- ↳ la mise en place de nouvelles méthodes de travail plus performantes et plus attentives aux besoins de chacun ;
- ↳ la mise en œuvre d'une nouvelle culture administrative visant à responsabiliser davantage ses collaborateurs.

La réussite de cette mission dépend de la capacité du Service Public Fédéral Finances à mobiliser ses ressources humaines autour de ces objectifs de qualité et de service, notamment par une valorisation permanente de ses fonctionnaires.

### **2.2.5. Conclusion**

En répondant aux attentes légitimes de la société, le Service Public Fédéral Finances veut renouer une relation de confiance avec ceux qui la composent, rétablir le consentement de l'impôt et s'assurer ainsi que chacun remplisse ses obligations dans le strict respect de la législation.



## **2.3. La réforme CoperFin : la rencontre des objectifs stratégiques de Copernic**

CoperFin est le nom donné aux projets de modernisation au sein du Service Public Fédéral Finances.

Ce nom évoque le lien initial avec la réforme Copernic (COPER), tout en soulignant l'appropriation des projets d'amélioration par les Finances (FIN).

Il indique aussi la volonté de placer les collaborateurs du Service Public Fédéral Finances au centre de la démarche entreprise

### **2.3.1. CoperFin : les BPR dans la pratique**

Le S.P.F. « Finances » a entamé une série de réformes draconiennes destinées à améliorer ses méthodes de travail, en mettant en œuvre pas moins de 16 programmes de BPR connus sous le nom de CoperFin.



**Figure 2-7 : Logo officiel CoperFin**

Onze de ces programmes sont spécifiques à certains domaines d'activités du S.P.F. Finances (douanes et accises, lutte contre la fraude fiscale, dossiers cadastraux, etc.). Cinq programmes généraux visent à améliorer les services fournis par le S.P.F., comme le programme de gestion des relations avec les citoyens, qui doit permettre de mieux comprendre ce que ceux-ci souhaitent et à adapter les services à ces besoins.

CoperFin est donc une refonte totale de la gestion des finances de l'État à travers laquelle tant la structure d'organisation que le fonctionnement du SPF Finances sont impactées.

### 2.3.2. Réforme organisationnelle

Dans le cadre de la réforme CoperFin, des groupes de travail ont été constitués et ont dessiné les schémas d'organisation des diverses entités sur la base d'un certain nombre de facteurs structurants, c.-à-d. d'éléments qui jouent un rôle important dans le dessin d'une structure. Ces éléments déterminent les processus ou parties de processus de travail et quelles fonctions sont regroupées dans une entité organisationnelle. Ces facteurs structurants ont été analysés pour chaque entité organisationnelle dans un Livre d'organisation.

Pour permettre un déroulement cohérent du développement du schéma d'organisation, un certain nombre de directives générales et un dessin de base avaient été élaborés à l'avance, sur lesquels les groupes de travail ont pu s'appuyer afin de déterminer les schémas d'organisation.

Les directives générales trouvaient essentiellement leur origine dans:

- ✦ Les principes Copernic, comme décrits dans la note au gouvernement intitulée "Fondements de la modernisation de l'administration fédérale"
- ✦ Le rapport final approuvé "Elaboration conceptuelle d'un nouvel organigramme pour le SPF Finances" (décembre 2000)
- ✦ Les principes stratégiques et le livre des processus développés dans le cadre de CoperFin.

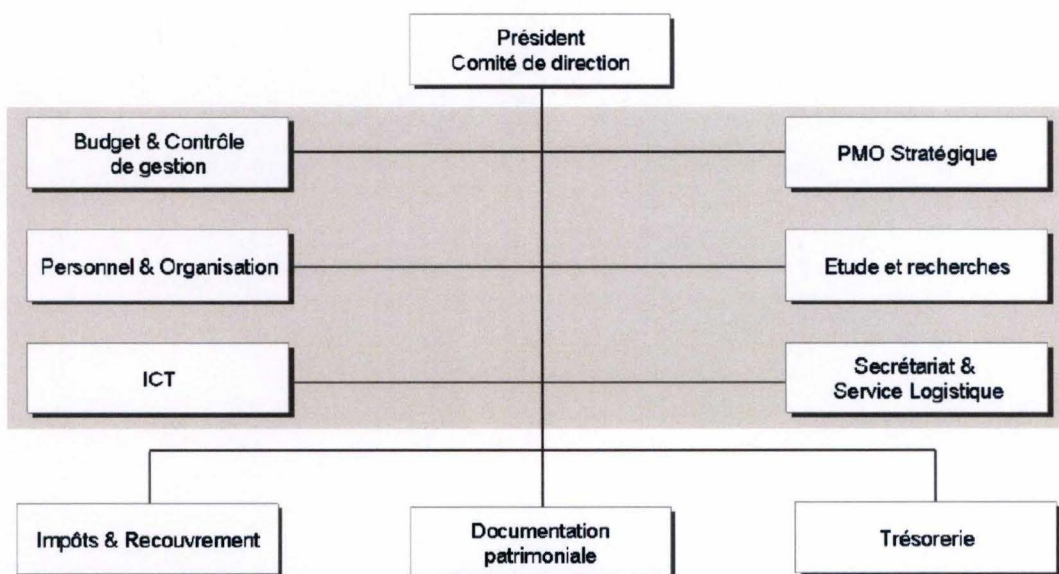


Figure 2-8 : Elaboration conceptuelle d'un nouvel organigramme pour le SPF Finances



Dans les “Fondements de la modernisation de l'administration fédérale”, il est fait surtout référence aux éléments relatifs à l'aplatissement de la structure de management et à la redéfinition de groupes de fonctionnaires:

- ↪ Une organisation qui multiplie les couches hiérarchiques freine la vitesse et handicape l'efficacité d'une organisation.
- ↪ Lorsque c'est possible, des mécanismes tels l'organisation de projet multidisciplinaire, l'élargissement des tâches ou encore le travail en équipe doivent remplacer le fonctionnement vertical existant en dépassant les limites des entités organisationnelles.
- ↪ Le nombre de couches et la taille des unités organisationnelles doivent être déterminés au cas par cas par le responsable de l'entité. Ils doivent pouvoir évoluer en permanence en fonction:
  - De la plus-value sociétale et stratégique des entités,
  - Du contenu, de l'ampleur et de la complexité de la matière traitée,
  - Du budget à gérer,
  - De la stratégie et du modèle d'organisation associé (par ex. Séparation entre divisions stratégiques et opérationnelles, centralisation ou décentralisation, système de contrôle interne, lignes de communication...),
  - De l'importance souhaitée des entités.
  - Du fait que l'objectif doit consister à réduire au maximum le nombre de couches hiérarchiques/dirigeantes, sans que la chose devienne ingérable pour les responsables.

De plus, la responsabilisation a été mise en exergue:

Une autonomie de gestion est dévolue à la direction de l'organisation. Cela signifie que le fonctionnaire dirigeant dispose d'une large autonomie sur le plan de l'organisation de l'entité dont il est responsable et de la manière dont il met en œuvre ses ressources pour accomplir sa mission.

Dans l'Elaboration conceptuelle d'un nouvel organigramme pour le SPF Finances, il est fait référence aux directives générales suivantes:

- ↪ L'interaction requise du contribuable avec le service public doit être aussi facile et limitée que possible (sur le plan de l'effort qu'il doit fournir et du nombre d'interactions). Elle doit être adaptée au maximum aux souhaits du contribuable, en tenant toujours compte des frais de gestion et des limites du cadre légal.
- ↪ Le traitement fiscal complet doit être un processus intégré au maximum. Cela signifie que le service proactif, le traitement des déclarations, les contrôles de gestion, le suivi des possibilités de recouvrement, la perception, le recouvrement et le traitement des contentieux y relatifs, doivent être intégrés au maximum. Cela se fait idéalement dans des postes de travail dans lesquels le traitement complet est assuré; sinon il faut veiller à un échange optimal des informations.
- ↪ L'entité qui s'occupe des contrôles approfondis doit effectuer ces contrôles pour tous les types d'impôts auxquels le contribuable est soumis.

## 2.4. Les conséquences de CoperFin

### 2.4.1. Volet processus : la refonte des processus de travail

Les processus du SPF Finances ont été regroupés en 7 grands groupes :

- ↗ **Processus fonctionnels Impôts et Recouvrements:** ces processus assurent le traitement en masse en ce qui concerne les déclarations et les dossiers des contribuables.
- ↗ **Processus fonctionnels Documentation Patrimoniale :** tous les dossiers parcourent ces processus.
- ↗ **Processus dirigeants:** ces processus assurent la sélection de certains dossiers, déclarations, contribuables, activités,... qui seront soumis à un traitement spécifique.
- ↗ **Processus de traitement spécifique:** ces processus assurent le traitement spécifique en ce qui concerne certaines sélections de déclarations, dossiers, contribuables,...
- ↗ **Processus de support:** ces processus fournissent le soutien nécessaire aux processus de traitement en masse, aux processus dirigeants et au processus de traitement spécifique.
- ↗ **Processus réglementation et méthodes de travail:** ces processus fournissent à l'organisation le soutien nécessaire dans le cadre du développement et de l'exécution de la législation.
- ↗ **Processus de management:** ces processus fournissent les directives nécessaires au fonctionnement de l'organisation.



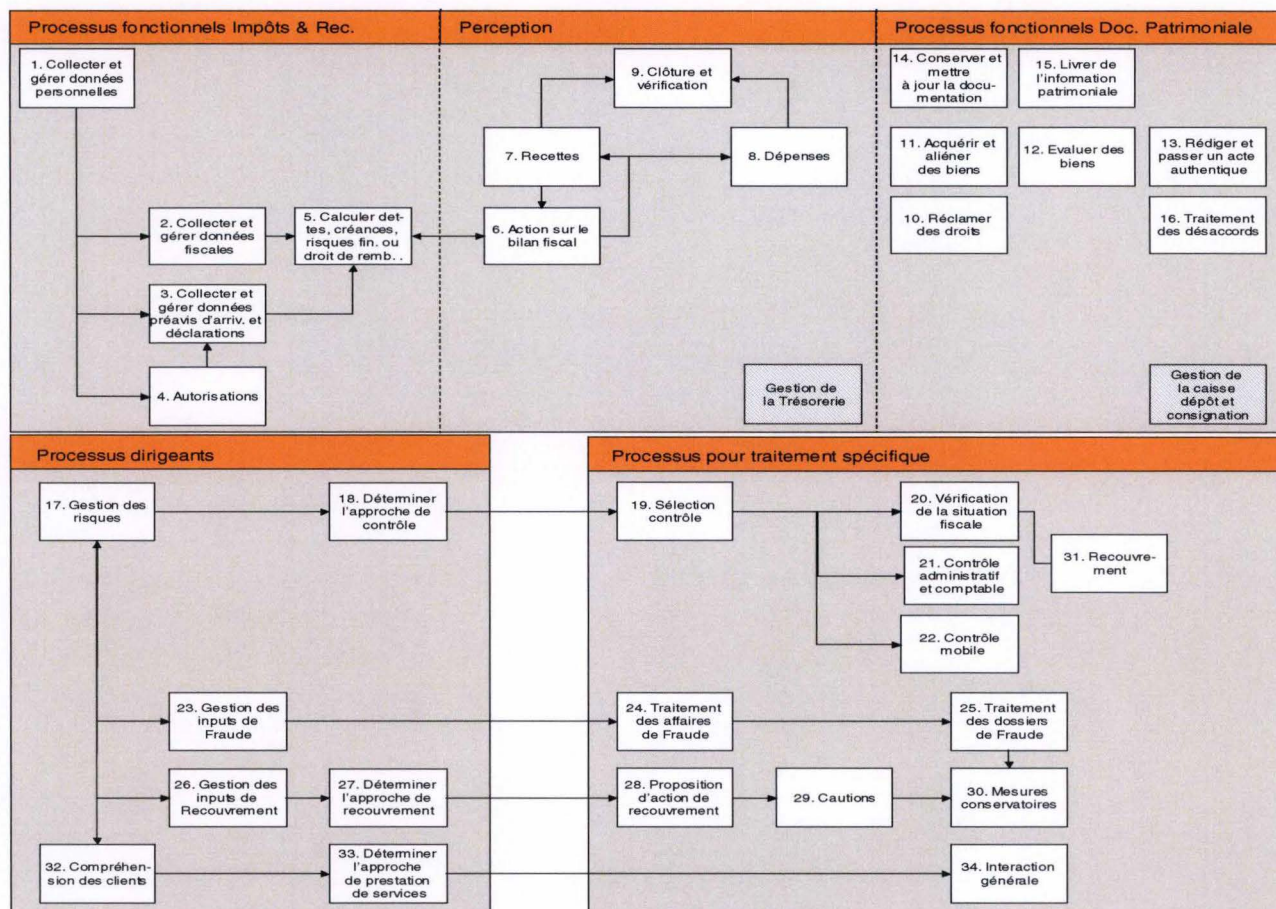


Figure 2-9 : Les nouveaux processus

Pour chaque processus, les éléments suivants ont été développés :

- ✎ La raison d'être ainsi que les principaux objectifs.
- ✎ La description des principaux blocs d'activités.
- ✎ Les déclencheurs, c'est-à-dire les événements qui initient le processus
- ✎ Les résultats, c'est-à-dire l'objectif final du processus, ce qui doit être générée.
- ✎ Les inputs, c'est-à-dire l'information qui est nécessaire pour exécuter un processus, une étape donnée.
- ✎ Les outputs, c'est-à-dire l'information qui est produite lors de l'exécution du processus
- ✎ L'Interface, c'est-à-dire les liens entre processus .

## 2.4.2. La modernisation du S.P.F. Finances : les aspects ICT

### • Les principes directeurs

Dans le cadre de la réforme CoperFin, le staff ICT a également fixé le cadre dans lequel le soutien informatique aux processus CoperFin serait apporté. Onze principes directeurs ont été retenus.

Par principe directeur, on entend un résumé des normes, standard et conventions (concernant la technologie, les processus et/ou l'organisation des services informatiques). Ils sont qualifiés de directeurs dans ce sens que les futures initiatives ICT devront satisfaire à ces directives si l'on désire réaliser les objectifs CoperFin proposés.

Il était presque impossible de définir des principes directeurs pour CoperFin sans convenir de quelques principes de base sur l'organisation informatique au sein du SPF Finances, lesquels pourraient ensuite être appliqués.

En effet, étant donné que le lien entre les entités métiers et les services informatiques est apparu comme l'un des principaux défis, il s'agit d'un facteur de succès critique pour la réussite des aspects ICT de CoperFin.

Le Réseau ICT a donc formulé 5 principes directeurs liés à l'organisation informatique au sein du SPF Finances :

#### 1. Mission du département informatique dans le SPF Finances :

- ⇒ Le département informatique est un partenaire commercial pour les autres départements du SPF Finances. Sa mission au sein du SPF Finances comprend deux volets :
  - L'informatique doit fonctionner en tant que facilitateur pour ses clients ; il offre des possibilités permettant au SPF Finances d'optimiser ses processus
  - L'informatique remplit son rôle de "fournisseur de services" ; il fournit donc des services pour garantir le bon fonctionnement du SPF Finances
- ⇒ L'informatique planifiera et exécutera toujours sa mission dans un contexte aussi large que possible. Il conviendra de tenir compte des projets locaux, départementaux et globaux au sein du SPF Finances. Les opportunités et les obligations seront également abordées dans un contexte plus large (ex. FEDICT, interactions avec les instances Européennes, ...).



## 2. Alignement Business – ICT :

- ↪ L'ICT remplit, avec le Business, un rôle important pour déterminer et réaliser les objectifs stratégiques du SPF Finances.
- ↪ Le Business et l'ICT sont donc conjointement responsables du planning et de la réalisation de projets au sein du SPF Finances.
- ↪ L'ICT veillera donc à ce que son personnel, ses processus et sa technologie contribuent à soutenir ces objectifs stratégiques.
- ↪ Pour ce faire, les projets devront satisfaire à un certain nombre de normes dans les domaines :
  - Organisation:  
Pour chaque projet, on formule des conventions claires en matière d'exigences et les conditions annexes (ex. budget, ressources, compétences) sont complétées par les parties concernées.
  - Processus:  
Chaque prestation de services de l'informatique sera définie à l'aide de SLA (Service Level Agreements) réciproques entre lui et l'entité métier concernée. Ils comprendront des normes de qualité, de reproductibilité, d'évolutivité, de gestion et de sécurité.
  - Technologie:  
La technologie requise doit pouvoir être adaptée à l'architecture ICT et être appuyée par un plan clair concernant la mise en phase et hors phase de cette technologie.

## 3. Interaction entre le Business – ICT – FEDICT :

- ↪ La responsabilité des discussions sur les aspects fonctionnels des processus professionnels est assumée par le Business et cela pour assurer :
  - La concertation interne : entre les différentes entités fonctionnelles du SPF Finances
  - La concertation externe : avec les autres SPF et instances.
  - En revanche, dans le cadre du re-engineering via les programmes BPR, chaque concertation concernant les discussions des aspects ICT des projets (potentiels) se déroulera avec les services informatiques.
- ↪ Via sa représentation au sein du comité de direction ICT permanent du FEDICT, le Réseau ICT assurera la coordination avec les autres SPF.
- ↪ L'informatique coordonnera son architecture technique au sein des services eux-mêmes. Il utilisera à cet effet les fondements technologiques disponibles du FEDICT et les développera avec sa collaboration. (volonté de standardisation).
- ↪ Le Business et l'ICT respecteront les services/fondements du FEDICT.

#### 4. Organisation de projets centralisées

- ↳ Les projets informatiques qui proviennent des programmes BPR, ainsi que les autres projets « métiers », seront décidés par la direction supérieure et suivis par les PMO (Program Management Offices) du SPF Finances (composés de représentants des entités métiers et de l'ICT). Ses objectifs sont :
- Garantir une intégration continue entre l'ICT et le Business.
  - Gérer le portefeuille de projets de l'ICT.
  - Définir les priorités.
  - Donner le maximum de chance de réalisation tant aux grands qu'aux petits projets
  - Garantir l'évaluation des initiatives qui dépassent le cadre du SPF.
  - Veiller à ce que les définitions de projets, les contrats de projets et leur gestion soient considérés comme des instruments de travail obligés

#### 5. Approche orientée projets :

- ↳ Tous les projets seront soutenus par le Business et l'ICT et seront définis sur la base de plans de projets clairs :
- Le Business déterminera la fonctionnalité souhaitée. Le Business sera le sponsor des applications à composante métier forte.
  - L'informatique déterminera la fonctionnalité des projets d'infrastructure ICT globale pour laquelle le SPF Finances mettra les moyens nécessaires à disposition.
  - L'informatique mettra en œuvre l'infrastructure et les applications. Elle veillera à la disponibilité des informations pour le Business.
  - Des accords seront conclus entre le Business et l'ICT sous la forme de contrats (Service Level Agreements, contrats de projets).
  - L'interaction entre le Business et l'ICT sera soutenue par des analystes professionnels.
  - Les rôles et responsabilités de chaque partie concernée seront définis au préalable.

Les 6 autres principes directeurs sont directement liés aux exigences en matière d'informatique dans une vision To-be, c'est à dire orientée vers la réalisation effective de la réforme Copernic.

#### 6. Architecture ICT

- ↳ L'environnement ICT du SPF Finances évoluera graduellement pour devenir une architecture stratifiée qui devra permettre une politique spécifique concernant :
- la gestion des informations
  - les applications Business
  - l'infrastructure ICT



↵ Cette architecture ICT est conçue autour de deux piliers :

- Des systèmes back-office intégrés assurent le traitement (transactionnel) des processus clés et sont exclusivement utilisés par les fonctionnaires du SPF Finances.
- Des applications front-office adaptées pour les utilisateurs internes et/ou externes qui sont axées sur le service, les informations et l'interaction avec les contribuables. On s'efforce à cet effet d'assurer une intégration efficace avec les systèmes back-office.

## 7. Normes ICT

- ↵ L'informatique du SPF Finances présentera une politique claire concernant les normes ICT qu'il utilisera.
- ↵ Ces normes doivent assurer une diffusion libre et rentable des informations et des processus.
- ↵ Dans ce contexte, le SPF Finances devra choisir autant que possible des normes "ouvertes".
- ↵ Les normes ICT évoluent, mais elles ont également un cycle de vie bien déterminé. Après cette période, elles seront remplacées sur la base des nouvelles normes.

## 8. Réutilisation de solutions informatiques dans un plus large contexte

- ↵ Le SPF Finances s'efforce de trouver des solutions ICT largement disponibles pour que les connaissances et l'expérience acquises puissent profiter à d'autres organisations.
- ↵ En revanche, le SPF Finances vérifiera si certaines solutions ICT ont déjà été réalisées dans d'autres organisations afin d'envisager une éventuelle.
- ↵ Le comité directeur ICT permanent du FEDICT est l'organe où de tels échanges peuvent intervenir.

## 9. Gestion des informations

- ✚ Les informations dans les applications du SPF Finances sont considérées comme un bien commun du SPF Finances et elles seront mises à la disposition des utilisateurs autorisés (internes et/ou externes).
- ✚ Cela sera concrétisé par l'attribution de responsabilités aux utilisateurs de ces informations (par l'attribution d'un rôle d'initiateur, par exemple, de créateur, de gestionnaire, de fournisseur et/ou d'utilisateur).
- ✚ Uniformité : Les entités et données structurées seront identifiées et normalisées de manière unique. Des conventions assureront l'intégrité et la cohérence des applications et des plates-formes. Le SPF Finances sera responsable de ce dictionnaire de données. Le département ICT y agira en tant que gestionnaire.
- ✚ Source authentique :
  - La collecte d'informations interviendra autant que possible de façon unique selon le principe de la source authentique. Cela signifie que l'origine (qui est de la plus haute importance pour l'exactitude des données) validera également ces données. Ces données seront alors utilisées au maximum.
  - En cas de modification de ces informations par d'autres sources, la source authentique doit en être avisée et devra avaliser celle-ci.
- ✚ Redondance :
  - Le stockage de données identiques (redondance) sera évité autant que possible et il ne pourra empêcher la recherche croisée des données.

## 10. Applications Business

- ✚ Lors de l'automatisation d'un processus professionnel (application) le SPF Finances considérera toujours les 2 options suivantes :
  - L'acquisition (et la personnalisation) de progiciels disponibles sur le marché ou mis à disposition par d'autres instances (ex. CE, ...). Le slogan : « ne fabriquez pas ce que vous pouvez acheter ».
  - La réalisation d'applications Business via un développement sur mesure.

## 11. Infrastructure ICT

- ✚ Une protection et une fiabilité cohérentes (disponibilité, évolutivité, etc.) de l'infrastructure informatique du SPF Finances constituent des conditions essentielles pour mettre les données du SPF Finances à la disposition des utilisateurs (instances, entreprises, citoyens).
- ✚ En outre, la collaboration entre les différents systèmes (actuels et futurs) du SPF Finances représente un "MUST". Elle devrait rendre possible l'intégration des applications et faciliter la mise en place de l'infrastructure future tout en permettant la continuation du day-to-day.
- ✚ On part du principe que les utilisateurs du SPF Finances disposent de postes de travail adaptés et normalisés (y compris au niveau des applications bureautiques) qui leur permettront d'accéder aux applications Business du SPF Finances.



### **2.4.3. La situation de départ : état des lieux des logiciels de travail existants**

Une analyse a évalué l'utilité des systèmes informatiques existants du SPF Finances pour les besoins des processus business CoperFin.

Il ne s'agissait pas d'une évaluation de la qualité et de l'efficacité des systèmes existants, ni d'une évaluation de l'utilisation de ses applications dans leur contexte de l'époque.

Dans le cadre l'analyse de l'existant, il a paru opportun de s'interroger sur une appréciation fonctionnelle et technique du système qui remplissait en ce moment les tâches en question (partiellement ou totalement conformes).

Cette analyse n'a donc certainement pas eu pour objectif de faire un inventaire complet de la situation de l'époque. Ce qui aurait eu peu de sens, puisque les processus « métiers » CoperFin annonçaient des modifications assez fondamentales.

L'analyse a eu principalement pour objet d'identifier les systèmes ou parties de systèmes susceptibles de faire partie de la future architecture informatique ; ce qui a d'ailleurs mis clairement en évidence l'écart (le 'fossé') entre la situation initiale et les besoins engendrés par la réalisation de CoperFin.

Comme déjà mentionné, l'analyse a été menée en fonction des besoins ICT futurs.

On s'est interrogé sur les points suivants :

↳ **Evaluation fonctionnelle:**

Evaluation qualitative des données et applications disponibles en fonction des fonctionnalités et tâches futures. Ceci a permis de voir dans quelle mesure les applications existantes pouvaient être réutilisées dans le contexte de la mission future.

↳ **Evaluation technique**

Evaluation qualitative de l'infrastructure disponible en fonction des fonctionnalités et tâches futures.

#### 2.4.4. L'adaptation des logiciels de travail suite à l'analyse de l'existant

Les applications actuelles sont constituées d'un grand nombre de logiciels non intégrées, historiquement développées en indépendance mutuelle et ne convenant guère pour supporter différentes nouveautés induites par les processus CoperFin, telles que :

- ✧ Le Dossier Unique (centralisation intégrée des informations fiscales et non fiscales),
- ✧ Le concept d'une "dette" unique (le bilan fiscal valable pour tous les impôts),
- ✧ Le principe fédéral de la source authentique (gestion des informations),
- ✧ Le système "intégré" qui englobe un traitement intégré de dossier transcendant les différentes entités concernées, quel que soit le type de transaction (impôt, prélèvement...),
- ✧ La gestion intégrée de toutes les interactions avec le "client", quel que soit le canal utilisé (échanges « papier », électroniques, verbaux, ...).

Ces principes fondamentaux induisent un re-engineering complet de ces applications fondamentales en soutien des processus CoperFin et en veillant à s'assurer que les connaissances fonctionnelles des systèmes existants soient reprises au maximum. De plus, un examen de l'impact qu'ont les principes directeurs informatiques sur la situation existante, a conforté la nécessité de réformer les systèmes actuels, et en particulier :

- ✧ La nécessité d'aller vers une architecture ICT à niveaux univoques, s'articulant autour de deux piliers (à savoir un back-office intégré et un front office).
- ✧ L'introduction d'une gestion centrale de tous les métas-data (dictionnaire des données) en soutien du principe de la source authentique pour les données fondamentales. Un autre élément à développer est la responsabilisation d'utilisateurs types des informations dans le but :
  - d'uniformiser et de standardiser les informations et leur gestion,
  - d'introduire le concept de source authentique,
  - d'éviter tout stockage redondant des données.
- ✧ S'assurer de la compatibilité entre les différents systèmes (existants ou nouveaux) au sein du SPF Finances
- ✧ Insérer les éléments technologiques constructifs des pouvoirs fédéraux (FEDICT) dans l'architecture du SPF Finances.

En ce qui concerne la réutilisation des systèmes existants dans un contexte futur, il est ressorti clairement que:

- ✧ Les applications fondamentales étaient riches en fonctionnalités mais répondaient insuffisamment aux besoins des processus futurs. Il fallait envisager un tout nouveau développement, tenant compte bien entendu des connaissances fonctionnelles et de l'expérience du SPF Finances.
- ✧ Les applications critiques actuelles devaient être progressivement remplacées par un système de traitement intégré, s'articulant autour du Dossier Unique.



## 2.5. Les bases IT du S.P.F. Finances

### 2.5.1. Contexte IT

Pour répondre aux objectifs stratégiques de la réforme CoperFin, le Service Public Fédéral Finances met donc en œuvre des projets innovants et de grande envergure qui constitueront les bases de sa nouvelle infrastructure informatique.

Ils s'articulent autour de plusieurs axes et visent notamment à mettre en place les fondements essentiels tant hardware que software de la future infrastructure informatique du département :

- ⇒ en fournissant les systèmes physiques de stockage et d'archivage standardisés utilisables par l'ensemble des applications et des systèmes du département (projet ATLAS)
- ⇒ en assurant la disponibilité et la fourniture des ressources matérielles nécessaires aux futures applications du département (projet ATLAS)
- ⇒ en apportant un système de gestion de bases de données relationnelles standard pour le département (projet RDC)
- ⇒ en mettant en place un service de gestion centralisé et uniformisé de la sécurité des accès pour l'ensemble des fonctionnaires du département (projet IDENTITY MANAGEMENT)
- ⇒ en mettant en place des lignes d'impressions de gros volumes pour les données sortantes des différentes applications mais dont les résultats imprimés devront également être visualisés via un browser Web (projet DOROTHEE).
- ⇒ en permettant à tous les agents du Service Public Fédéral des Finances d'accéder via leur browser Web à un système centralisé de gestion électronique de documents acquis par numérisation ou de documents électroniques tirés de lignes d'impressions de gros volumes (projet DOMINIQUE)

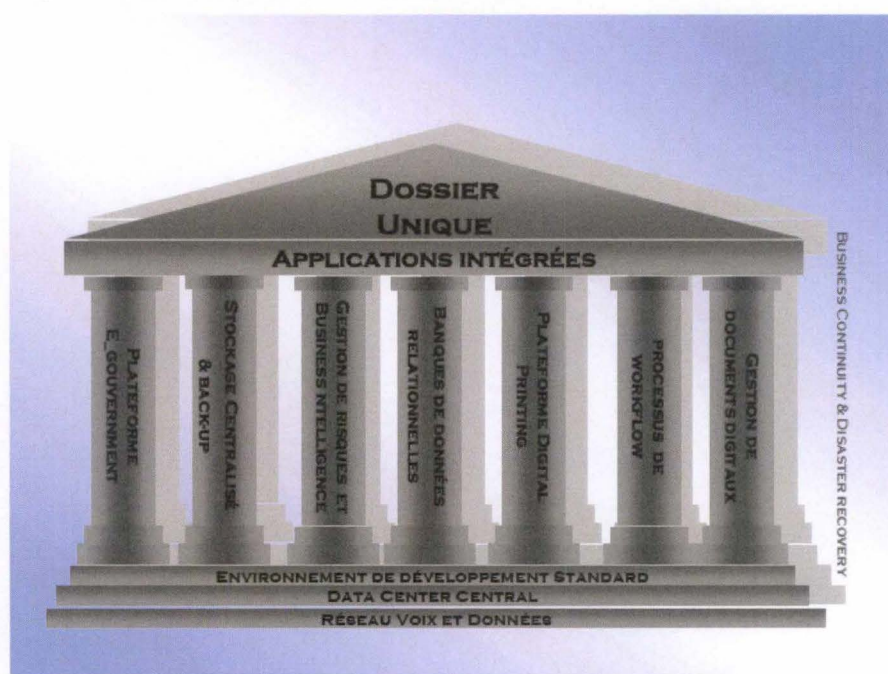


Figure 2-10 : Les fondements ICT et «métier» du S.P.F. Finances

Ces différents fondements devront permettre de :

- ↳ répondre aux besoins internes du Département
- ↳ réaliser l'intégration interne et externe des systèmes d'informations
- ↳ préparer les moyens à la mise en place d'un futur Customer Relationship Management
- ↳ rencontrer les demandes des citoyens.

### **2.5.2. Cadre spécifique**

Si, depuis les projets de passage à l'an 2000 et passage à l'Euro, le concept de projet et les techniques de gestion de projet sont devenues familières aux acteurs des projets informatiques du S.P.F. Finances, on peut constater que la définition des différentes phases d'un projet, des deliverables qu'elles doivent produire ainsi que les rôles joués par les différents intervenants varient de projet à projet, en fonction des partenaires externes ou des différents administrations. concernées.

Cette situation a une conséquence importante.

Les intervenants «métier» et les professionnels de l'informatique du S.P.F. doivent, pour chaque projet, s'adapter aux méthodes de travail de leurs partenaires. Ceci a pour effet de compliquer la collaboration entre les différents acteurs et de la rendre moins efficace. Cette constatation peut également s'appliquer aux projets internes.



### **3. Introduction d'une méthodologie de développement : le projet Support au Développement (SupDev)**

#### **3.1. Les enjeux**

L'introduction d'une méthodologie de développement d'applications devra permettre de formaliser et de stabiliser le déroulement des projets dans un cadre commun en uniformisant :

- ↗ le contenu des étapes de ceux-ci
- ↗ les rôles des différents acteurs
- ↗ les livrables de chaque étape.

Ceci permettra à l'intervenant « métier » et ICT d'avoir une vue complète du déroulement d'un projet de pouvoir à tout moment situer l'état actuel du projet par rapport aux objectifs à atteindre.

Si l'introduction d'une méthodologie de développement d'applications permet de définir tâches et rôles dans un projet, elle suppose :

- ↗ l'introduction d'un « langage » commun entre les différents intervenants
- ↗ la définition d'un ensemble de règles formelles, de diagrammes et de conventions auxquels les documentations et livraisons du projet devront répondre et ceci au cours des différentes étapes du développement.

Ces règles formelles devront permettre de traduire et de formaliser les informations fournies par l'intervenant « métier » en informations permettant la construction d'applications ICT.

Vu la complexité croissante de ces applications, la formalisation nécessaire est de plus en plus complexe. Cette complexité la rend quasi impossible à réaliser sans le support d'outils spécialisés permettant au minimum :

- ↗ la réalisation de diagrammes complexes consistants et corrects
- ↗ le partage et la consolidation de ceux-ci dans un projet et au travers des différents projets de l'entreprise
- ↗ la construction d'un dictionnaire de données commun à l'entreprise.
- ↗ un support à la transformation des diagrammes en code exécutable

Ces outils permettent également de faciliter la conception, la validation et la gestion de bases de données complexes comme celles qui seront hébergées par le RDC (Relation Data Center). Ils permettent de plus le reverse engineering des bases de données existantes et une aide au reverse engineering des sources de données telles que fichiers indexés séquentiels ...

En résumé, la mise en place d'une méthodologie de développement d'applications informatiques permettra de créer un environnement dans lequel « métier » et ICT pourront collaborer d'une manière transparente et « répétable », tandis que le déploiement d'outils de modélisation permettra aux intervenants « métier » de fournir de manière structurée l'information fonctionnelle requise et aux intervenants ICT de l'exploiter rigoureusement en vue de réaliser les applications ICT répondant aux besoins exprimés.

## **3.2. Les objectifs du projet**

Le projet SupDev a pour but l'implémentation d'une méthodologie et d'outils de support au développement au S.P.F. Finances.

Ce projet crucial pour l'organisation a fait l'objet d'un cahier des charges précis et le marché a été attribué à une société externe afin de nous accompagner dans la mise en œuvre de la méthodologie et dans l'implémentation logicielle des outils de maintenance et de développement.

### **3.2.1. Mettre en place un standard imposé à tous les intervenants**

La plupart des nouveaux projets de développement informatique au sein du S.P.F. Finances se font via le recours à des sociétés externes.

La mise en place d'une méthodologie de développement devra assurer l'uniformisation des méthodes et des résultats attendus afin d'assurer la standardisation et la facilité de maintenance.

La méthodologie devra s'appuyer sur un ensemble de standards, de référentiels et de prescriptions techniques ou méthodologiques qui sont en place ou en cours de définition au S.P.F. Finances. Ces standards ont été définis pour permettre de maintenir la cohérence globale du système d'Information.

Nous citons ci-dessous certains de ces standards et expliquons quel rôle ils joueront au sein de la méthodologie.

**CCFF** décrit l'architecture de référence des applications J2EE et fournit l'ensemble des composants du serveur d'application J2EE pour pouvoir permettre le développement des applications. Dans le cadre de la méthodologie, CCFF est en quelque sorte le fondement de l'approche de construction par assemblage de composants.

**RDC** vise la mise en commun et la rationalisation des données «métier» et définit au sein de la discipline Analyse/Conception la conduite à suivre pour tout ce qui concerne l'accès aux données.

Le **BPR CoperFin** définit le cadre métier et fonctionnel TO-BE. Il sera utilisé comme ligne directrice de la phase d'Inception pour définir le cadre métier dans lequel les applications seront développées.



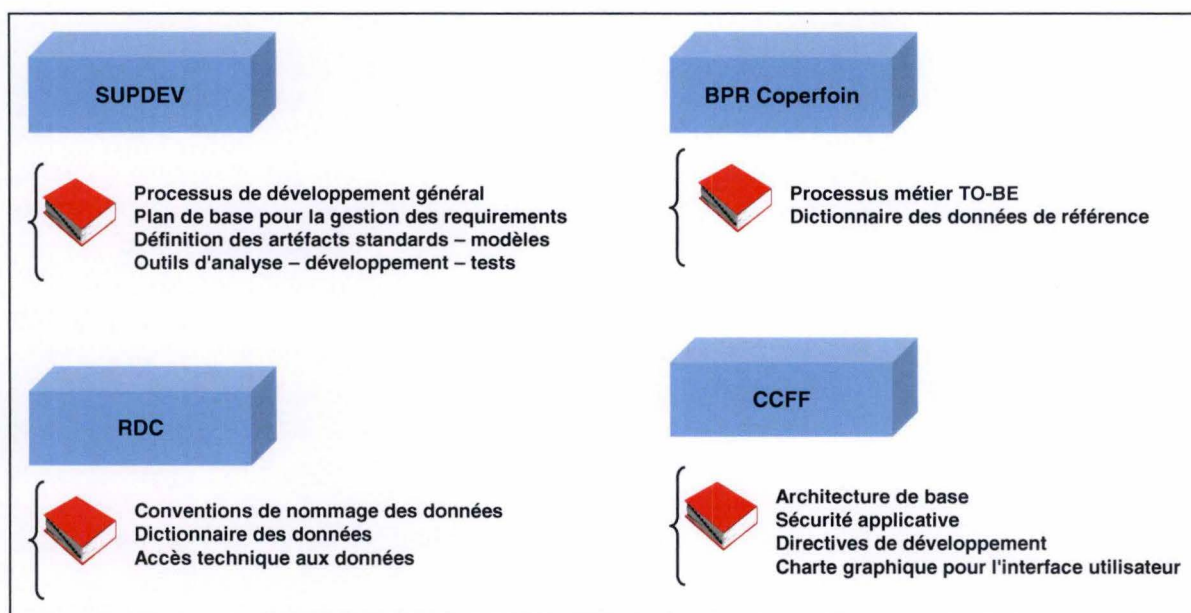


Figure 3-1 : Les différents standards au S.P.F. Finances

La prise de connaissance de ces référentiels est hautement souhaitée lors de la définition de toute nouvelle application ou de toute modification majeure d'une application existante.

### 3.2.2. Maîtriser les coûts de développement

L'uniformisation des développements informatiques devra assurer une meilleure maîtrise des coûts des projets et éviter les dépassements de ceux-ci.

### 3.2.3. Mettre en place un cadre strict et porteur

Le besoin d'un cadre strict, dûment défini et balisé, devra permettre à l'organisation de maîtriser toute la chaîne de développement de ses nouvelles applications informatiques.

### 3.2.4. Assurer la polyvalence

La méthodologie à mettre en place devra être capable de couvrir tous les types de développement prévus et à prévoir au sein du S.P.F. Finances.

Que ce soit au niveau de la taille du projet, de son budget ou des ressources qui y sont affectées. La méthodologie devra aussi s'appliquer tant aux projets purement internes qu'aux projets externalisés.

### 3.2.5. Maîtriser toutes les étapes et les productions

La méthodologie devra assurer la maîtrise de toutes les étapes essentielles du cycle de vie d'une application informatique, à savoir :

- ↗ La gestion des spécifications
- ↗ L'analyse conceptuelle
- ↗ Le design technique
- ↗ La modélisation de données dans un environnement précis
- ↗ La programmation
- ↗ L'élaboration des tests
- ↗ Le déploiement
- ↗ La gestion d'un environnement de travail collaboratif, la gestion des versions ...

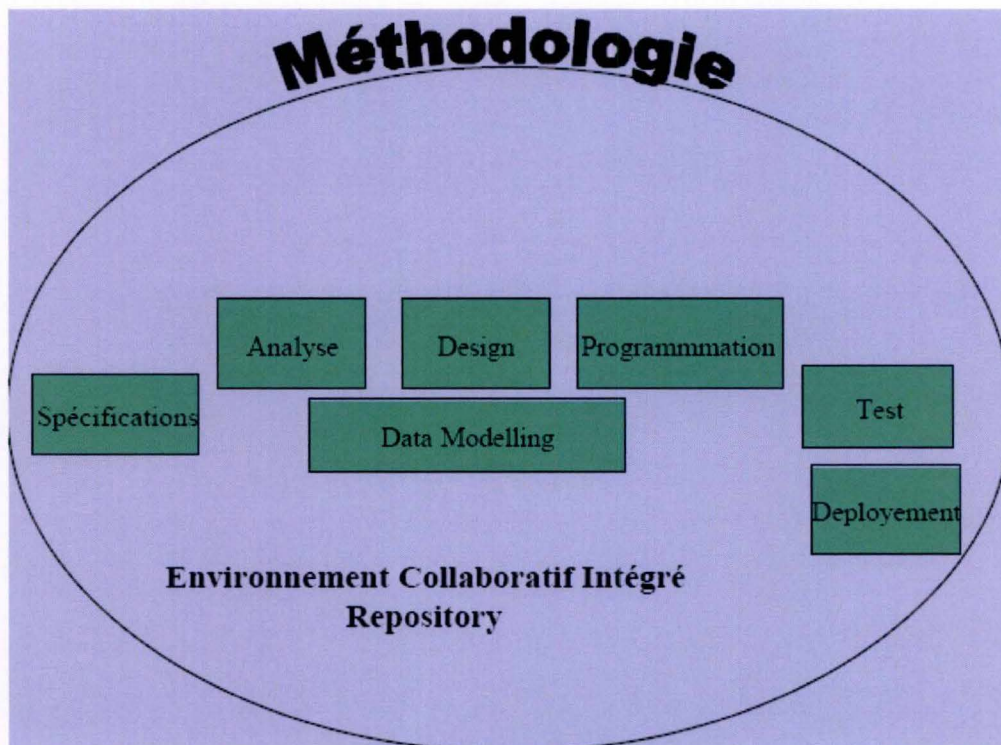


Figure 3-2 : La couverture méthodologique



### **3.2.6. Assurer la prise en compte des aspects « métier » du développement**

Trop souvent les développements informatiques sont confinés dans les services de programmation qui ne maîtrisent pas suffisamment les implications « métier » de leur développement.

La mise en place de la méthodologie devra assurer la prise en compte de l'analyse « métier » des projets en assurant la captation et la gestion des spécifications.

### **3.2.7. Assurer la maîtrise d'une suite logicielle permettant le soutien de la méthodologie par l'élaboration de modes opératoires précis**

Les outils de support au développement auront pour but de couvrir toutes les étapes du cycle de vie des applications définies dans la méthodologie.

Les logiciels seront utilisés par les analystes d'application, application designer, développeurs, testeurs et administrateurs de bases de données du S.P.F., ainsi que par les analystes et développeurs externes participant aux projets du S.P.F.

La méthodologie définie devra leur permettre d'utiliser les outils choisis dans un cadre compatible avec les principes définis au cours des BPR CoperFin.

### **3.2.8. Assurer une évolution qualitative des développements**

La formalisation des livrables et l'utilisation rationnelle et optimale des outils de développement devra amener une évolution qualitative substantielle des développements.

### **3.2.9. Assurer l'évolution qualitative de l'organisation**

Bien que le but initial de la mise en place d'une méthodologie de développement ne soit pas directement liée au souhait de faire évoluer l'organisation vers un degré de maturité reconnu, il est essentiel d'assurer l'évolution de cette dernière vers des critères de qualité mesurables et évolutifs.

### **3.2.10. Mettre en place la centralisation des informations**

Afin de couvrir les besoins « métier » et d'en assurer la bonne compréhension, la méthodologie devra mettre en place un glossaire transversal à tous les projets qui devra être mis à jour en permanence.

Quant au dictionnaire des données et des services, lui aussi transversal, il devra assurer la rationalisation des concepts et des données y associées. Il assurera également la réutilisabilité des services déjà développés.

### **3.3. Le périmètre du projet SupDev**

#### **3.3.1. Dans le périmètre du projet**

- **Méthodologie :**

La mise en place d'une méthodologie de développement d'applications informatiques devra permettre de créer un environnement dans lequel Métier et ICT pourront collaborer d'une manière transparente, structurée et « répétable ».

- **Outils de développement :**

Le déploiement d'outils de modélisation devra permettre aux intervenants « métiers » de fournir de manière structurée l'information fonctionnelle requise et aux intervenants ICT de l'exploiter rigoureusement en vue de réaliser les applications ICT répondant aux besoins exprimés.

Ces outils permettront également de faciliter la conception, la validation et la gestion de bases de données complexes comme celles qui seront hébergées par le RDC. Ils permettent de plus le reverse engineering des bases de données existantes et une aide au reverse engineering des sources de données telles que fichiers indexés séquentiels ...

- **Formations :**

Mise en place de formations devant permettre aux premières équipes de développement du S.P.F. de mettre en œuvre les outils et méthodologie proposés ; d'autre part des formations devant permettre à une cellule du service de staff ICT d'assurer la formation et le support des utilisateurs internes du S.P.F..

- **Gestion du changement**

Outre la formation des différents intervenants aux développements d'un projet informatique, SupDev devra prendre en charge la gestion du basculement méthodologique des projets « nouvelles technologies » en cours de développement au sein du S.P.F. Finances et leur assurer aide et assistance.

- **Support technique des outils de développement**

Ceci comprend non seulement l'aide à l'utilisation des outils, maintenance des serveurs d'application mais aussi basculement des modélisations en cours sur d'autres outils vers les outils SupDev pour les outils « nouvelles technologie » en cours de développement au sein du S.P.F. Finances.

#### **3.3.2. Hors du périmètre du projet**

- ✧ L'étude de faisabilité d'un projet ;
- ✧ La maintenance des applications en production ;
- ✧ La gestion d'un projet (budget, ressource, planning, ...)
- ✧ L'aspect de communication interdépartementale ;



## **3.4. Les critères de réussite**

### **3.4.1. Mettre en place un standard imposé à tous les intervenants**

Actuellement les projets de développement informatique diffèrent beaucoup dans leur approche méthodologique et dans la qualité de réalisation. Ils dépendent fortement des différents intervenants impliqués et de leur maîtrise des processus.

Depuis plusieurs années, les services ICT s'efforcent de mettre en place des standards aussi bien en matière de matériel qu'en matière d'outils de travail.

Le projet SupDev a pour but premier de mettre en place un standard méthodologique appuyé par une suite logicielle qui devra répondre à ces besoins.

La volonté est clairement affichée d'imposer ce standard à tous les intervenants aux projets ; intervenants qui sont aussi demandeurs d'un cadre strict et éprouvé qui permettra d'assurer une stabilité des connaissances et une efficacité accrue.

### **3.4.2. Maîtriser les coûts de développement**

La maîtrise des coûts de développement au sein d'un S.P.F. n'est pas chose aisée dans la mesure où la majorité des projets d'investissement sont attribués via des appels d'offre à des sociétés externes.

Ces marchés sont déterminés à prix fixe par le soumissionnaire en fonction du périmètre du projet.

La conséquence directe est qu'en cas de découverte d'un élargissement du périmètre du projet, le financement de celui-ci n'est pas assuré par l'enveloppe budgétaire octroyée au projet. On doit alors procéder à une nouvelle demande de crédit via un « change request » (demande de changement) qui devra être soumis à l'aval du gouvernement.

Ces élargissements de périmètre peuvent être la conséquence de plusieurs causes :

- ↳ Le périmètre initial du projet a changé entre la soumission et la réalisation ;
- ↳ Les dispositions légales ont été modifiées depuis entre la soumission et la réalisation du projet ;
- ↳ Les ressources internes nécessaires à la réalisation du projet ont mal été définies, ce qui entraîne une charge de travail supplémentaire pour le soumissionnaire, en termes de ressource et/ou de temps ;
- ↳ Les interdépendances entre projets ont mal été définies ;
- ↳ Un manque de complétude dans la définition des périmètres de projets, souvent remarqués lors de la consolidation de ceux-ci.

Le souci de nos managers de mieux maîtriser les coûts de développement des projets informatiques par la mise en place d'une méthodologie et d'outils de développement est un facteur de succès pour le projet SupDev. En effet, la mise en place de procédures uniformisées, la

centralisation des informations dans les outils, la rationalisation des développements et la mise en commun des informations devaient résoudre la plupart des manquements constatés ci-avant.

### **3.4.3. Mettre en place un cadre strict et porteur**

Comme nous le verrons dans la section « Contraintes », la prise en compte de la mentalité du personnel du S.P.F. Finances et des méthodes de travail en cours depuis des années sera déterminante.

Les fonctionnaires ont l'habitude de travailler dans un cadre strict, gouverné par des procédures de travail contraignantes et précises. Ce mode de gestion organisationnelle a montré son utilité dans la gestion du travail de plus de 30.000 fonctionnaires.

C'est d'ailleurs dans le même schéma de pensée que les nouveaux processus de travail CoperFin ont été travaillés et que la refonte des fonctions est envisagée.

Dans ce cadre de réforme CoperFin, la mise en place d'une méthodologie de développement informatique est grandement facilitée et supportée par la refonte des fonctions. En effet, des fonctions particulières pourront être mises en place pour répondre au mieux aux besoins de la méthodologie.

Les procédures de travail seront assurées par les différents livrables méthodologiques ainsi que par les modes opératoires des outils assurant la transposition entre la méthodologie et les outils.

### **3.4.4. Assurer la polyvalence**

Le fait de mettre en place une méthodologie qui permettra de gérer tous les projets informatique du S.P.F. Finances, quelles que soient sa taille et sa complexité sera un atout majeur pour l'adhésion à celle-ci.

Un souci constant d'adaptation devra être apporté tout au long du développement de cette méthodologie afin d'adapter celle-ci en fonction :

- ✧ Des critères de complexité et de taille qui définissent un projet ;
- ✧ Des différents types de projets envisageables (internes, externes) ;
- ✧ Des livrables obligatoires et les livrables « optionnels » en fonction du projet en cours ;
- ✧ Des ressources affectées au projet ;
- ✧ Du périmètre du projet (nouvelles applications, adaptations d'applications existantes, ...)

Assurer la polyvalence de la méthodologie permettra aux différents intervenants d'acquérir une maîtrise et une expertise nécessaires à l'évolution qualitative des développements et de l'Organisation. Elle devra répondre aux attentes des fonctionnaires et des intervenants externes impliqués.



### **3.4.5. Maîtriser toutes les étapes et les productions**

Le besoin de spécialisation des différents intervenants dans un projet informatique est une demande très souvent relayée par les membres du personnel.

La mise en place d'une méthodologie axée sur les rôles et fonctions, délimitant un cadre strict des interventions de chaque acteur et spécialisant ceux-ci est une réponse optimale aux attentes.

L'adhésion du personnel est donc grandement facilitée mais ne pourra être acquise que si l'organisation met en place une gestion des ressources humaines digne de sa politique. (Voir « Risques »).

### **3.4.6. Assurer la prise en compte des aspects « métier » du développement**

Suite à la réforme CoperFin et à la mise en place des nouveaux processus de travail, la dimension « métier » est au centre de toutes les préoccupations.

La prise en compte des besoins « métier » et des spécifications de ceux-ci est enfin comprise et acceptée par les services ICT du S.P.F. Finances qui très longtemps se sont trouvés isolés des préoccupations « métier » pour ne développer que des applications n'y répondant que partiellement.

Il s'agissait avant tout d'un problème de mentalité qui au travers du temps et du renouvellement du personnel de mieux en mieux formé aux techniques de développement actuelles disparaît.

La fonction d'analyste « métier » est apparue depuis quelques années et rencontre une reconnaissance de plus en plus grande auprès des services ICT.

La formalisation de cette fonction à travers la méthodologie permettra une meilleure intégration des dimensions métiers et ICT.

### **3.4.7. Assurer la maîtrise d'une suite logicielle permettant le soutien de la méthodologie par l'élaboration de modes opératoires précis**

Le recours aux méthodes de modélisation moderne soutenues par des outils informatiques est un aspect du développement assez nouveau au S.P.F. Finance. Les concepts de centralisation des données de modélisation, de traçabilité et de réutilisabilité sont eux aussi assez nouveaux.

L'utilisation de nouvelles techniques soutenant la méthodologie n'est plus à démontrer et la réalisation de quelques projets (souvent des préétudes) avec ces techniques a convaincu de l'utilité d'y recourir.

L'absence de standard logiciel en la matière au sein de l'organisation et le passage à des technologies de programmation différentes (de Cobol à Java) sont des facilitateurs à la mise en place d'une suite logicielle soutenant la méthodologie.

Les modes opératoires précis et les sessions de formation aux outils assureront la maîtrise du changement des méthodes de travail.

### **3.4.8. Assurer une évolution qualitative des développements**

Les conséquences positives apportées par la mise en place d'un cadre méthodologique guidant et fonctionnel, par l'installation d'une suite logicielle performante et par la formation de nos agents, devront répondre aux besoins de recherche de qualité imposés par la hiérarchie.

Dans le cadre de la réalisation de ce projet, nos décideurs ont mis sur pied un projet de « Quality Control » assuré par une firme extérieure qualifiée en la matière.

La collaboration étroite entre ce projet et le nôtre permettra de nous assurer l'aval de la hiérarchie et légitimera auprès de tous les intervenants des projets, tant internes qu'externes, la qualité et la recherche d'efficacité de notre démarche.

Le résultat final de notre projet (méthodologie et suite logicielle), soutenu par les interventions du « Quality Control » ne pourra qu'amener à une évolution qualitative des développements.

### **3.4.9. Assurer l'évolution qualitative de l'organisation**

Bien que l'objectif principal de la mise en œuvre d'une méthodologie de développement ne soit pas l'évolution qualitative de l'organisation, elle y contribue cependant.

Mais elle ne suffit pas. En effet, pour pouvoir parvenir à assurer la maturité qualitative de l'organisation, il faut en plus prendre en compte l'aspect « gestion de projet » et l'aspect « gestion de l'organisation ».

Le premier aspect sera pris en compte par la mise sur pied d'une méthodologie de gestion de projet efficace et standardisée au sein du S.P.F. Finances. Cet aspect est en cours de réalisation dans un autre projet.

L'objet de ce programme ne faisant pas partie de ce mémoire, nous ne tracerons ici que les grandes lignes de son périmètre.

De manière assez parallèle à SupDev, ce projet a pour but d'implémenter la méthodologie Prince 2 (méthodologie de gestion de projet mise au point en Angleterre par les organismes étatiques) et soutenu par la suite logicielle NIKU.

Le second aspect, la « gestion de l'organisation » est prise en compte par la mise en œuvre au sein de notre organisation de PMO (Program Management Office) qui ont pour but de soutenir nos managers dans la gestion des projets et programmes de leur entité. Ce soutien est aussi bien axé sur l'axe stratégique (aide à la décision, orientation politique, élaboration de plan stratégique annuel ...) que sur l'axe opérationnel (gestion des dépendances interprojets, gestion des ressources inter départementales, ...).

La volonté managériale d'établir une chaîne complète, allant de la décision à la réalisation, nous assure le soutien de nos décideurs dans la réalisation de notre projet.



## **3.5. Risques**

### **3.5.1. Risques liés au contexte**

- ↙ Impact organisationnel
  - => Délai de mise en place des nouvelles structures
- ↙ Impact culturel
  - => Acceptation par le personnel des réformes engagées
- ↙ Impact technologique
  - => Maîtrise des nouvelles technologies
- ↙ Impact politique
  - => Poursuite de la réforme en cas de changement de majorité gouvernementale

### **3.5.2. Risques liées à l'ICT**

- ↙ Changements organisationnels
  - => Mise en place des nouvelles structures
- ↙ Changement de plateforme technologique
  - => Maîtrise des nouvelles technologies, helpdesk
- ↙ Formalisation du processus de développement
  - => Acceptation des nouveaux processus de développement liés aux nouvelles technologies
- ↙ Disponibilité des ressources
  - => Maintenance et développements conjoints dans la nouvelle et l'ancienne structure
- ↙ Compétence des ressources
  - => Volonté de formation des ressources impliquées

## **3.6. Facteurs critiques de succès**

- ↙ Soutien du Management et de la chaîne hiérarchique
- ↙ Adhésion du personnel
- ↙ Planning réaliste tenant compte d'une phase de test (projets pilotes).
- ↙ Engagement et compétence des experts méthodologiques

### 3.7. Hypothèses de départ

Nous énonçons ici les hypothèses au moment de la proposition du projet et qui doivent se réaliser pour assurer la réussite de ce dernier.

- ✧ Soutien du management ;
- ✧ Enveloppe budgétaire suffisante ;
- ✧ Mise en place d'une cellule de management du changement ;
- ✧ Suivi du service du personnel pour la refonte des rôles et fonctions ;
- ✧ Mise en place organisationnelle des nouvelles structures de développement de projet informatique ;
- ✧ Collaboration entre le métier et l'ICT ;
- ✧ Mise à disposition des ressources humaines pour le groupe de travail, appelé « core team » ;
- ✧ Mise à disposition des ressources techniques (gestion du réseau, serveurs, renforcement en mémoire des ordinateurs de développement, ...) ;
- ✧ Recrutement de personnel devant assurer la formation des fonctionnaires selon le principe du « Former les formateurs » ;
- ✧ Déterminer un ou deux projets pilotes afin de tester les solutions définies.



### 3.8. Les contraintes

#### 3.8.1. Organisation du service IT du S.P.F. Finances

L'organisation de l'ICT au S.P.F. Finance est en plein changement. Le modèle ancien était structuré par les différentes administrations qui détenaient chacune ses propres départements ICT.

Depuis septembre 2004, l'ensemble des départements ICT ont été regroupés pour ne former plus qu'un seul département, lui-même organisé comme suit :

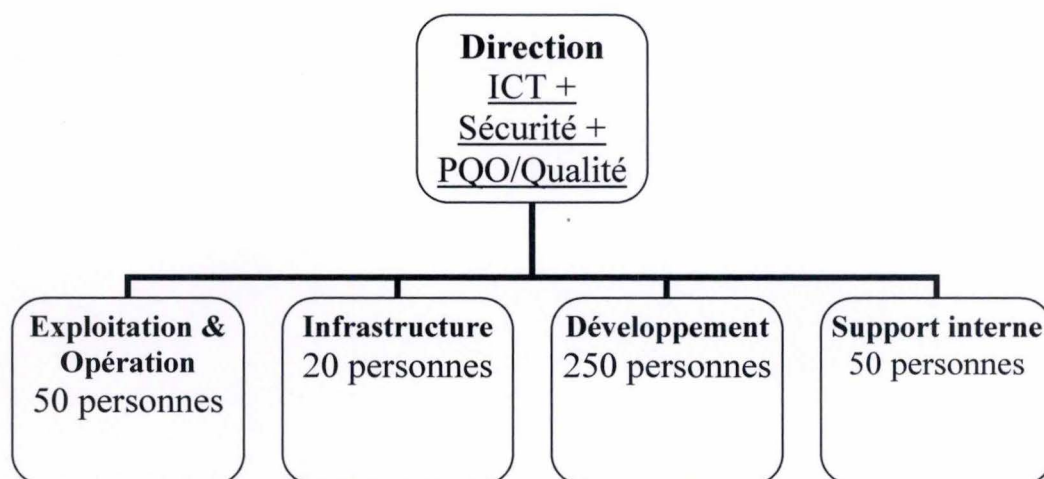


Figure 3-3: Organisation du service ICT.

Organisation	Rôle
Exploitation	Gestion des opérations informatiques en termes de mise à disposition opérationnelle des systèmes informatiques
Infrastructure	Responsable de la définition des standards en termes de logiciels, architectures et méthodologies. Faire des choix qui permettent une pérennité.
Développement	En charge du développement applicatif de nouveaux projets informatiques et de la maintenance des systèmes existants.
Support interne	Support interne ou help desk bureautique interne au S.P.F. Finance.

Il y a donc actuellement un total d'environ 370 personnes à l'ICT. CoperFin avait estimé ce besoin à 800 personnes !

Le manque de ressources est donc un point d'attention critique.

#### 3.8.2. Organisation du service d'analyse métier du S.P.F. Finances

Les analystes « métier » actuellement en fonction au S.P.F. Finances sont regroupé au sein de plusieurs services et ont pour tâche principale la maintenance et l'évolution des applications existantes. De ce fait, ils assurent plus le « day to day » que les analyses nécessaires au développement de nouvelles applications.

Le faible nombre de ces ressources et la quantité de travail qui leur est assigné posent le grave problème de la disponibilité de ces ressources.

Cet aspect constitue une contrainte forte pour la phase de travail du « core team » qui a pour but de définir et de développer la méthodologie.

Ce fait constitue aussi un risque non négligeable au moment de l'implémentation de la méthodologie (Voir Risques).

### **3.8.3. Les différents types de projets et leur organisation**

Il existe principalement 3 types de projet :

- ↳ les projets internes s'appliquant souvent aux systèmes anciens (soit les projets « legacy »),
- ↳ les projets internes bureautiques ou informatiques
- ↳ les projets externes utilisant souvent les nouvelles technologies.

Remarquons que jusqu'en 2002, il n'y avait pratiquement pas de projets externes, et sauf quelques exceptions, tous les développements étant internes !

Dans tous les cas, l'analyse « métier » et l'expression des besoins seront réalisées par les départements « administratifs » (métiers) sauf si le projet est à dominance technique ou à usage interne de l'ICT.

Les trois grands domaines administratifs du S.P.F. sont évidemment encore d'application. En simplifiant on pourrait les structurer comme suit : Patrimoine (Immeuble, Cadastre), Taxation et recouvrement (TVA, Contributions directes, etc.) et Secrétariat généraux (Ressources humaines, etc.).

#### **Les projets internes**

Les projets internes en rapport avec les systèmes « legacy » sont définis, réalisés, et pilotés sur tout le cycle de développement sans l'intervention de participants externes. Ils sont souvent réalisés par une équipe de 1 à 10 personnes au maximum.

Dans ce type de projet, la plupart des participants connaissent le métier et les besoins. Ceci a pour conséquence que l'analyse des besoins et l'analyse fonctionnelle sont souvent tributaires de l'expérience et du degré de formation des intervenants au projet. De ce fait, elles peuvent être de différent format et taille allant depuis une description très succincte jusqu'au descriptif complet mais probablement textuel.

Les développements sont majoritairement en COBOL, langage prédominant au S.P.F. Finances à cause de son environnement Mainframe, standardisé jusqu'il y a peu.

La méthodologie de développement qui supporte ces projets est traditionnelle, liée aux systèmes propriétaires. Elle tente d'intégrer par contre des aspects itératifs et s'appuie sur une certaine réutilisation.



Exemple :

Pour construire les calculs du précompte professionnel d'un nouvel exercice d'imposition, on réutilise la version antérieure que l'on adapte aux nouvelles spécifications. De plus, un calcul d'exercice d'imposition d'un exercice d'imposition donné doit toujours rester accessible et exécutable même plusieurs années plus tard avec les mêmes règles de calculs (même s'il y a eu des améliorations) étant donné que l'on doit toujours pouvoir refaire des calculs sur des périodes antécédentes. L'historisation et la nécessité de maintenir les versions successives pour pouvoir retraiter des dossiers dans le contexte où ils ont été initialisés sont donc importantes. Les versions successives d'une même application sont donc amenées à devoir vivre très longtemps et doivent rester mobilisables à tout instant pour retraiter un dossier dans son contexte originel.

La gestion de ces projets se fait sur base d'une gestion de « change request » au moyen d'un outil de développement dans la maison.

Ce type de projet n'est en principe pas à couvrir par le projet SUPDEV bien qu'il soit souhaité que la méthodologie puisse être d'application pour la définition des interfaces entre les systèmes « legacy » et les nouveaux systèmes, ainsi que pour la gestion des documentations, des applications, des données et des tests.

### **Les projets internes bureautiques**

Les projets internes bureautiques font références aux projets PC ou projet intranet.

Dans ces 2 cas, l'équipe de projet se résume souvent à une très petite équipe et la méthodologie utilisée est non publiée. En effet, dans ce cas l'équipe (1 ou 2 développeurs) connaît le métier et les besoins et livre une solution sur base de cette connaissance ou de spécifications partiellement structurées.

Ce type de projet n'est pas à couvrir par le projet SUPDEV bien que la méthodologie proposée puisse être également suivie par ce type de projet si elle est adaptable à l'importance du projet.

### **Les projets externes**

Les projets externes sont structurés en une phase de préétude suivie d'une phase d'appel d'offre publique qui au terme d'un processus de sélection attribue le marché de réalisation à un fournisseur qui effectuera avec le S.P.F. Finances la phase de développement.

La phase de préétude comprend l'analyse de faisabilité, la définition du business case et l'écriture du cahier des charges pour la phase de développement.

La phase de développement est la partie qui comprend certainement le développement de la solution, les tests et la mise en production de celle-ci.

Il est important de noter que les fournisseurs voulant travailler avec le S.P.F. Finances seront contraints d'adopter les standards définis par le projet SUPDEV pour le pilotage et la réalisation des projets informatiques. Ces standards définiront aussi la stratégie de développement des applications informatiques.

L'adoption de standards vise à garantir une meilleure maîtrise du patrimoine informatique, un transfert de connaissance et une appropriation plus aisée.

Les développements spécifiques en dehors de ces standards seront donc à l'avenir tout à fait exclus.

#### A. Phase de Préétude

Cette phase comprend l'analyse de faisabilité, la définition du business case et l'écriture du cahier des charges. La principale livraison de cette phase consiste donc en un cahier des charges en français et en néerlandais.

Une fois ce cahier des charges réalisé, un appel d'offre est lancé pour la phase de développement de l'application.

Actuellement les cahiers de charges peuvent être de différent formats et contenus allant depuis le simple texte descriptif des besoins (analyse fonctionnelle) jusqu'à une approche plus orientée « use case » avec une description de chacun de ceux-ci. Il s'agit dans ce cas d'une description métier que la solution développée devra supporter.

Il est à noter qu'actuellement il semble qu'il n'y a pas beaucoup de synergie entre les différents projets et que dans certains cas, il arrive que l'on reconstruise ce qui a déjà été construit dans un projet connexe. Il semble donc qu'une meilleure consultation des projets connexes ainsi qu'un référencement à des parties d'autres projets dès l'écriture du cahier des charges est indispensable au bon fonctionnement du S.P.F. Finances.

#### B. Appel d'offre publique

La phase de préétude est suivie d'un processus officiel d'ouverture de marché public qui comprend :

- ↳ Une publication du cahier des charges ;
- ↳ La réception et l'ouverture des offres ;
- ↳ Une séance publique de questions réponses ;
- ↳ Une évaluation des offres par une commission d'évaluation qui comprend des parties représentants le métier et des parties représentants l'ICT ;
- ↳ Une attribution finale du marché.

La transition depuis la livraison du cahier des charges jusqu'au démarrage de projet de développement est typiquement de 1 an !

Il est important de noter que les fournisseurs de la phase de développement ne peuvent pas être les fournisseurs de la phase de préétude.



### C. Phase de Développement

La phase de développement est activée par le kick-off qui officialise le début du projet. Cette phase est effectuée dans une relation client-fournisseur sur le principe du schéma du maître d'ouvrage – maître d'œuvre.

Cette phase devrait en principe réutiliser ce qui a été construit dans la phase de préétude.

On constate cependant que dans beaucoup de cas, l'étude des besoins est incomplète ou a évolué : elle a eu lieu il y souvent plus d'un an.

Un projet est souvent formé d'un groupe de 30 à 50 personnes, de plus en plus souvent organisé par un binôme de chef de projet:

- ↳ Un chef de projet « métier » représentant les intérêts de la MOA. Il est responsable d'obtenir tous les livrables d'un point de vue métier ;
- ↳ Un chef de projet « technique » assurant le pilotage du projet au quotidien, le suivi de l'avancement et la qualité vis-à-vis des fournisseurs.

La spécialisation des rôles, l'importance de la prise en compte du métier dans l'analyse et le contenu fonctionnel souvent très lourd des nouvelles applications poussent à des configurations avec 2 chefs de projets.

La tendance est de constituer des équipes mixtes client/fournisseur sur la phase de réalisation, ceci de manière à faciliter les transferts de connaissance entre client et fournisseur. L'objectif étant à moyen terme de constituer un véritable centre interne de compétences, maîtrisant les nouvelles technologies afin d'être capable de gérer toutes les problématiques liées à un nouvel environnement technologique.

Les tests et l'exécution des tests sont en général exécutés par les gens « métier ».

Les projets de développement du S.P.F. Finances sont caractérisés par le découpage « préétude » / « projet » :

- ↳ Les préétudes ou 'avant-projets'

Les préétudes sont réalisées par le S.P.F. Finances et/ou via des sous-traitances externes.

Dans ce cas, cette préétude fait elle-même l'objet d'un cahier des charges spécifique.

Si le développement se fait avec un fournisseur, elle se solde par un cahier des charges, prétexte à un appel d'offres.

- ↳ Le projet lui-même = projet de réalisation

Il s'agit du projet de développement de l'application informatique. Souvent ces projets font l'objet d'un contrat avec un fournisseur.

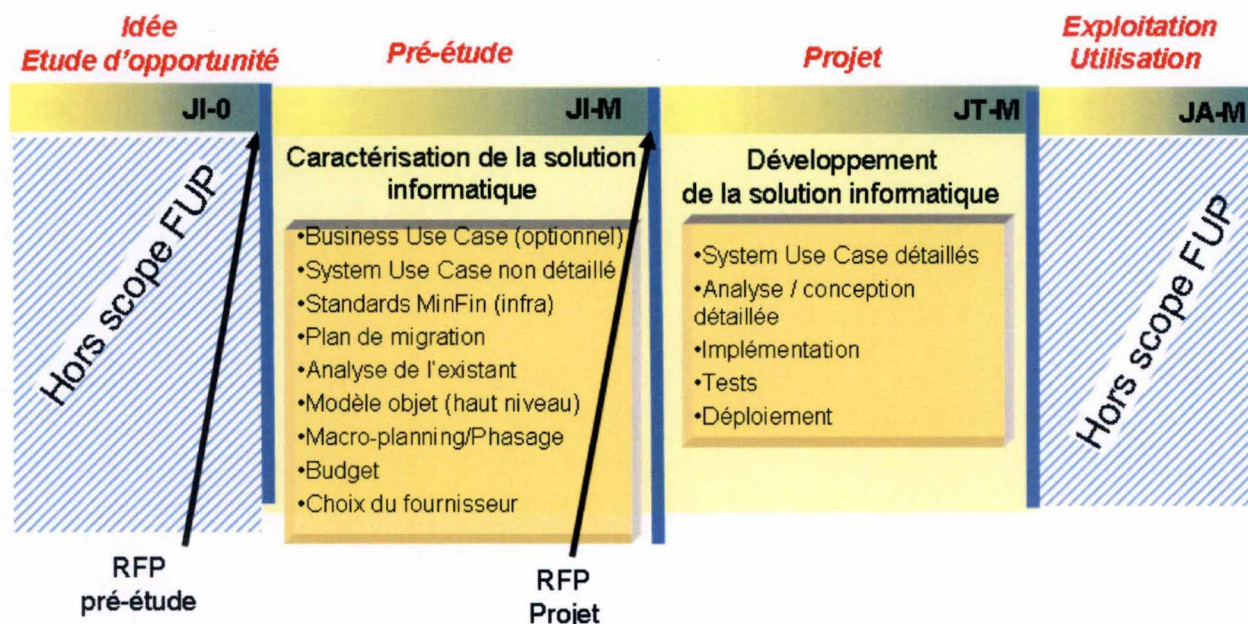


Figure 3-4: Découpage « préétude » / « projet »

➤ Conclusion :

Le projet SupDev sera un projet externalisé sans phase de préétude. Un cahier des charges sera réalisé en interne et soumis à appel d'offre. Le projet sera donc tributaire des offres remises et du choix méthodologique exprimé dans celles-ci.

### 3.8.4. Mentalité du personnel d'un S.P.F.

Comme nous l'avons déjà signalé plus avant, le service ICT du S.P.F. Finances a longtemps été isolé et considéré comme un service à part. En effet, durant des années, les seules applications informatiques présentes au ministère étaient uniquement axées sur la délivrance d'informations et l'encodage de celles-ci, sans tenir compte des processus de travail ni des spécificités « métier » des différents intervenants.

Ceci est sans doute dû à l'environnement « mainframe » en place à l'époque, qui de par sa lourdeur était fort statique et peu adaptable aux demandes naissantes du métier.

L'arrivée des nouvelles technologies de développement, du monde PC et de l'intranet a bouleversé ce petit monde qui s'est trouvé obligé de s'adapter de force à ces changements.

Les traditionnels programmeurs en Cobol durent alors côtoyer les deux mondes, ce qui ne manqua pas de poser beaucoup de problèmes.

Les managers de l'époque décidèrent alors de former sur base volontaire les programmeurs et les analystes à ces nouvelles technologies. Mais il est vite apparu que cela ne suffirait pas sans recourir également au recrutement de chefs de projets et d'analystes formés à ces techniques.

Le renforcement du service ICT par l'arrivée de nouvelles ressources, souvent fraîchement sortie des études et sans expérience solide, eut pour effet de voir émerger une multitude de



méthodes de travail différentes et souvent non coordonnées. L'efficacité recherchée n'était pas atteinte.

Le recours à des procédures de travail dûment décrites et formalisées mit un peu d'ordre dans les développements. Cependant, elles furent établies en internes par des personnes qui n'étaient pas des spécialistes en méthodologies ; et quoique répondant à un besoin de guidance, elles ne prirent en compte que les relations entre sous-services sans mettre au centre des préoccupations le projet.

L'expérience acquise lors de la réalisation de projets basés sur les nouvelles technologies et l'expertise métier des différents intervenants (administrateurs réseaux, administrateurs de bases de données, programmeurs, analyste fonctionnels, architectes, ...) firent ressortir les besoins et contraintes suivantes :

Besoins	Contraintes
Définir des procédures de travail directives	Axer les procédures sur l'aspect métier
Définir une méthodologie axée sur le projet	Coordonner les différents métiers et sous-services

### **3.8.5. Restructuration organisationnelle et fonctionnelle**

La réforme CoperFin en cours au S.P.F. Finances bien qu'initiatrice de ce projet de mise en place d'une méthodologie de développement, est aussi source de contraintes pour l'implémentation de celle-ci. La refonte des processus métiers et la mise en place des nouvelles fonctions doivent être un préalable à la mise en place de la méthodologie.

En effet, la définition des rôles et fonctions, ainsi que celle des processus de gestion de projet et de mise en place des PMO doit être avalisée par le service P&O (Personnel et Organisation) afin de pouvoir fonctionner dans le nouvel environnement.

### **3.8.6. Fonctionnement par appel à des sociétés externes**

Le fait d'avoir recours à des sociétés externes pour la réalisation de projets informatiques pose souvent le problème de l'intégration et de l'uniformisation des développements. Chaque intervenant externe possède bien souvent ses méthodes de travail propres et les impose au S.P.F. Finances.

La mise en place d'une méthodologie standardisée dans notre organisation, contraignante pour tous les intervenants, devrait améliorer cet état de fait.

L'intégration de ce nouveau standard dans les cahiers de charge sortant à partir de 2006 ne pose pas de problème puisque l'acceptation des standards du S.P.F. est un préalable à la prise en compte des offres. De plus, les sociétés externes ont pu intégrer dans leurs offres les coûts d'un tel standard, l'adaptation des ressources en fonction de celui-ci et l'intégration du partage des informations et la planification du projet.

Pour ce qui est des projets en cours de développement, l'intégration du standard méthodologique n'est pas sans conséquences directes. La réorganisation des ressources, l'adoption en plein développement de nouveaux outils de modélisation et de programmation, le besoin de formation à ces nouvelles données de travail ont un coût non négligeable.

Ces coûts ne sont pas uniquement financiers (hausse des budgets de +/- 15%), mais également temporels. Nous remarquons ainsi que l'adoption en cours de route du nouveau standard méthodologique provoque une extension du temps de développement de un à six mois en fonction de l'état d'avancement du projet.

Cependant, la hausse de qualité des développements, l'intégration des différentes applications dans les outils de modalisation, la centralisation des concepts, des services et des données ainsi que la complétude des analyses doivent nous amener à un rapport coût/bénéfice positif.

### **3.8.7. Intégration de l'approche métier et ICT**

Comme nous l'avons signalé plus tôt dans ce travail, l'intégration de la dimension métier et technique ICT n'était pas de tradition au S.P.F. Finances. De ce fait, il persiste au sein des différents services un état d'esprit de méfiance et de cloisonnement.

Les services ICT se plaignent des spécifications « métier » trop souvent volatiles et incomplètes, et les services d'analyses fonctionnelles de restrictions trop fortes de la part de l'ICT. En fait, il s'agit surtout d'une incompréhension persistante dans la manière de travailler en commun dans un but commun.

L'implémentation de la méthodologie devra tenir compte de cet état d'esprit et tenter d'y remédier.

## **3.9. Liens avec d'autres projets**

Une définition d'un standard interne orienté vers la qualité du développement et la gestion managériale des programmes et projets est actuellement en cours d'étude et semble s'orienter vers une approche PMO (Program Management Office) avec des tableaux de bords, des indicateurs, KPI, gestion de planning, gestion budgétaire, etc. ... et une approche Prince 2 pour la gestion de projet.

Le projet SUPDEV étant davantage centré sur la méthodologie de développement, il est fort probable que les projets « PMO » et « Gestion de Projet » influenceront à terme fortement la méthodologie de développement.



### 3.10. Démarche

#### 3.10.1. Définition du processus de sélection

- Définition de l'orientation méthodologique

Rapidement il est apparu que le besoin d'adapter une méthodologie préexistante aux besoins et contraintes du S.P.F. Finances était le meilleur moyen d'amener notre organisation à un niveau de compétence et d'efficacité optimal.

Partir d'une méthodologie existante et éprouvée offre l'avantage de profiter de la maturité et de l'expérience acquise dans le domaine.

#### Approche classique versus approche orientée objet (OO)

Actuellement, deux approches méthodologiques du développement de logiciels de gestion sont en concurrence. L'approche fonctionnelle ou classique, datant des années 1970, est issue de la programmation structurée ; l'approche orientée objets, datant des années 1990, est issue de la technologie de programmation orientée objets.

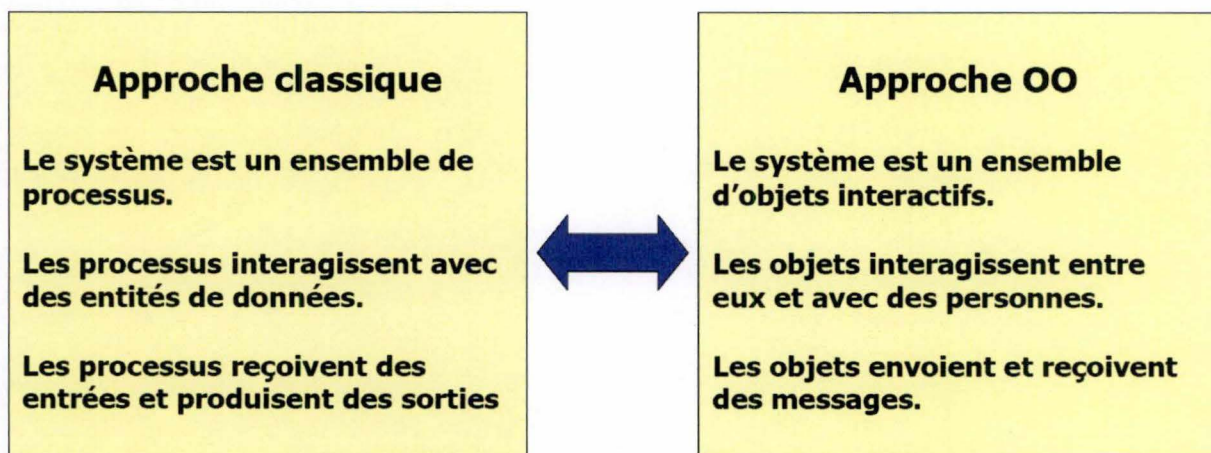


Figure 3-5 : Approche classique et orientée objet

Les paradigmes de la programmation structurée et de la technologie orientée objets étant drastiquement divergents, il existe une forte propension à opposer les méthodes issues de ces deux courants de pensée.

A l'opposé, s'agissant de la compréhension du métier, de la définition des exigences ou de la conduite du projet de développement, la technologie utilisée pour l'implémentation n'a que peu ou pas d'incidence sur la démarche méthodologique.

Les pratiques et méthodes contemporaines de développement de logiciels, quelle que soit l'approche, préconisent toutes des activités de modélisation métier, de gestion des exigences, d'analyse, de conception, d'implémentation, etc.

Ces activités permettent de passer de concepts intellectuels au code exécutable par niveaux d'abstraction successifs.

En fait, les deux approches méthodologiques divergent pour les activités produisant les livrables les plus concrets et elles convergent pour les activités produisant les livrables les plus abstraits.

Par livrables les plus concrets, il faut comprendre l'implémentation sous forme de code ou d'architecture et par livrables les plus abstraits, il faut comprendre la gestion des exigences, l'analyse et la conception.

Par ailleurs, les contraintes de délai, de réactivité, de satisfaction des utilisateurs et autres imposées aux analystes, concepteurs ou programmeurs ne sont pas dépendantes d'un choix méthodologique de développement de logiciel mais sont le fait d'exigences des commanditaires.

Quel que soit l'environnement d'implémentation choisi, on devra souvent pratiquer un développement itératif et incrémental, la gestion des risques ou encore le maquettage et le prototypage pour satisfaire aux attentes du client.

Les deux approches ne sont pas séparées par un fossé infranchissable; elles ont de nombreux points communs en planification et conduite de projets mais aussi en expression des besoins et en analyse.

Toutefois, elles s'éloignent en phase de conception, voire divergent en phase d'implémentation.

Pour le choix d'une méthodologie, il est évident qu'il faut retenir une approche orientée objets si l'implémentation est réalisée à l'aide d'une technologie orientée objets; si l'implémentation est réalisée par l'intermédiaire de programmation structurée, il faut retenir une approche classique ou fonctionnelle.

L'adéquation des natures de méthodologie et d'environnement d'implémentation, classique ou fonctionnelle d'une part et orientée objets d'autre part, permet un gain d'efficacité grâce à l'homogénéité des concepts du couple méthode et langage de programmation.

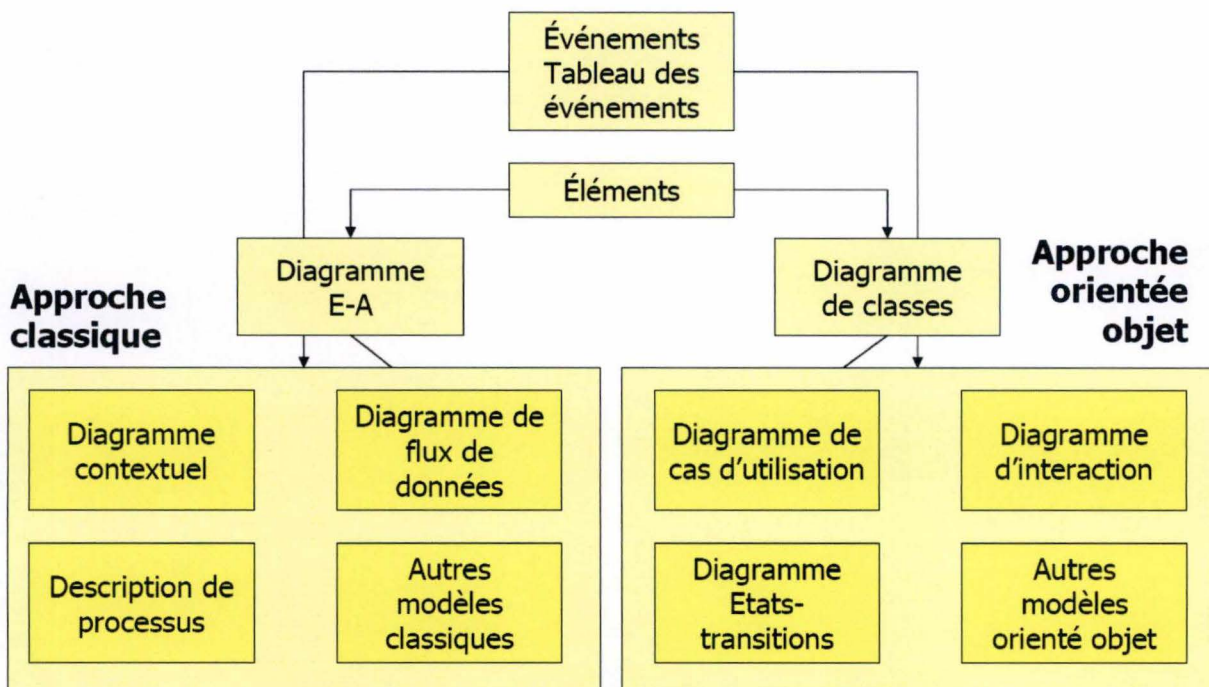


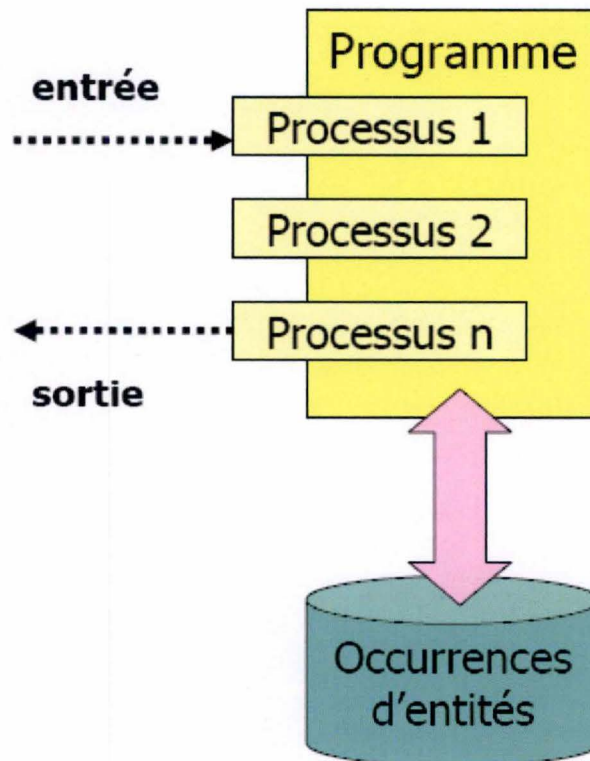
Figure 3-6 : Les principaux livrables selon l'approche



### **L'approche classique ou fonctionnelle : la donnée au centre du système**

L'approche classique s'axe sur les techniques utilisées pour développer des systèmes d'information au moyen de la programmation structurée et modulaire.

Ces techniques qui aident le développeur à définir ce que le système doit faire (les spécifications de traitement), les données que le système doit stocker et utiliser (les spécifications de données), quelles entrées et sorties sont requises et comment les fonctions (les processus de traitement des données) travaillent ensemble pour exécuter des tâches.



**Figure 3-7 : L'approche classique – données – processus/programme**

Dans un premier temps, la modélisation des processus permet de comprendre le métier et les activités à mener dans le système. Dans un deuxième temps, la modélisation des flux de données permet de comprendre ou spécifier le fonctionnement du système d'information.

L'importance ainsi octroyée aux données résulte d'une hypothèse clé selon laquelle les données forment une ressource organisationnelle qui risque de changer bien peu comparativement aux processus de l'entreprise. Ce sont les besoins en données internes et externes qui sont censés déterminer les processus et la structure des processus, et non l'inverse.

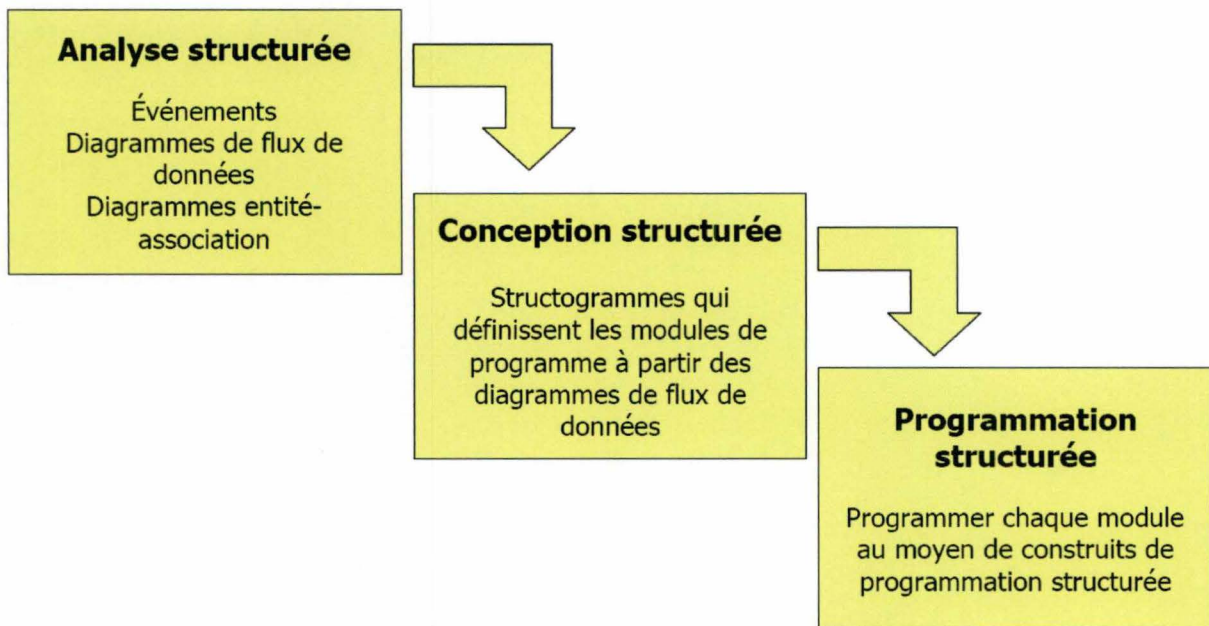


Figure 3-8 : De l'analyse structurée à la programmation structurée

Du fait de mettre ainsi les données au centre du développement, la dynamique du programme ne s'axe que sur ces dernières et force ainsi l'organisation à adapter ses processus de travail à la maintenance de celles-ci.

### L'approche orientée objet : la méthode au centre du système

La principale différence entre l'approche orientée objets et l'approche classique est que les objets s'occupent aussi bien du traitement de l'information que de son stockage.

L'objet est une unité atomique formée de l'union d'un état et d'un comportement. Il fournit une relation d'encapsulation qui assure à la fois une cohésion interne très forte et un faible couplage avec l'extérieur.

L'objet révèle son vrai rôle et sa vraie responsabilité lorsque, par l'intermédiaire de l'envoi de messages, il s'insère dans un scénario de communication et de traitement des tâches à accomplir.

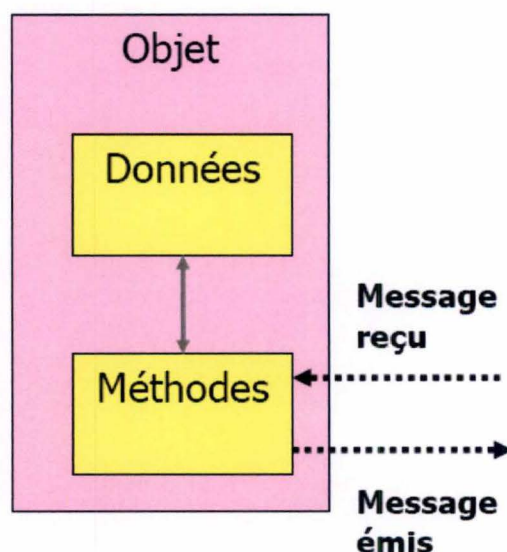


Figure 3-9 : La dynamique de l'objet



Cela est possible parce que les objets possèdent non seulement des attributs mais aussi des méthodes. Les modèles de spécification de l'approche orientée objets finissent par se démarquer considérablement de ceux de l'approche classique à cause du comportement des objets. Quant aux modèles de conception, ils sont tout à fait différents.

L'approche orientée objets considère un système d'information comme une collection d'objets interdépendants qui fonctionnent de concert pour exécuter des tâches. Il n'y a ni processus ni programmes, ni entités de données ni fichiers dans le système, mais uniquement des objets. Un objet est un élément du système qui est capable de répondre à des messages,

Dans une organisation, la distinction des composants va de pair avec leur intégration. La performance de l'approche objet provient alors de sa double capacité à décomposer –différencier – et à recomposer –réunir –du fait de la richesse de ses mécanismes d'intégration, qui concernent à la fois les aspects statiques et dynamiques des logiciels.

L'approche objet tire sa force de sa capacité à regrouper ce qui a été séparé, à construire le complexe à partir de l'élémentaire, et surtout à intégrer statiquement et dynamiquement les constituants d'un système.

## Les méthodes Agiles

Les méthodes agiles sont des méthodes de gestion de projets informatiques qui tentent de trancher avec les méthodes traditionnelles de développement jugées trop lourdes.

Ces méthodes sont regroupées au sein de l'Agile Alliance et elles obéissent à quatre grandes valeurs:

1. Priorité aux personnes et aux interactions par rapport aux procédures et aux outils  
=> travail en groupe, communication
2. Priorité aux applications fonctionnelles par rapport à une documentation pléthorique  
=> Documentation succincte à jour, documentation permanente du code
3. Priorité à la collaboration avec le client par rapport à la négociation de contrat  
=> Rapport régulier au client, solution répondant réellement aux attentes  
=> Grande maturité du client, relation de confiance
4. Priorité de l'acceptation du changement par rapport à la planification  
=> Planning flexible, modifications possibles après une 1<sup>ère</sup> version du système

Ces quatre valeurs sont déclinées sur des principes généraux :

- ✧ Satisfaire le client en lui livrant très tôt et régulièrement des versions fonctionnelles de l'application  
=> le client peut décider de la mise en production de l'application
- ✧ Accueillir les demandes de changement, même tard dans le processus de développement  
=> produire des systèmes flexibles
- ✧ Livrer le plus souvent possible des versions opérationnelles de l'application avec une fréquence temporelle la plus courte possible  
=> le but est de livrer une application qui satisfasse aux besoins du client

- ☞ Instaurer une collaboration quotidiennement entre le client et le développeur tout au long du projet
- ☞ Construire des projets autour d'individus motivés à qui on fait confiance pour remplir leur mission
- ☞ Indiquer l'avancement du projet par le fonctionnement de l'application
- ☞ Instaurer un rythme soutenable au projet et maintenir ce rythme constant  
=> Adapter le rythme pour préserver la qualité du travail sur la durée du projet
- ☞ Porter une attention continue à l'excellence technique et à la conception  
=> maintenir le code source propre, clair et robuste
- ☞ Simplifier est essentiel
- ☞ Permettre à l'équipe de s'autoorganiser
- ☞ S'interroger à intervalles de temps régulier sur la manière de devenir plus efficace et ajuster les comportements en conséquence.

Pour certains, les méthodes traditionnelles et les méthodes agiles sont complémentaires. Les méthodes agiles ne seraient que la mise en œuvre du bon sens et du pragmatisme des développeurs qui utilisent les méthodes traditionnelles.

### **Aperçu des principales méthodes agiles :**

#### **ASD (Adaptive Software Development),**

Méthode par laquelle le processus de développement est en continuelle adaptation selon le principe de la répétition de cycles d'expérimentation, de collaboration et d'apprentissage. Ces cycles remplacent le traditionnel processus de développement en cascade.

Cette méthode est surtout recommandée pour la gestion des changements et sa capacité d'adaptation au cas par cas, mais reste difficilement applicable pour les projets importants demandant des efforts de développement relativement longs.

#### **Scrum,**

Scrum est basée sur le concept que le développement logiciel n'est pas un processus défini, mais une démarche empirique. Démarche empirique selon laquelle la force est dans le groupe et dans sa capacité à réagir rapidement dès qu'une faiblesse est perçue sans recourir à la planification. Elle est basée sur des processus de travail implicites.

Cette méthode requiert de maintenir une bonne communication, d'anticiper les conflits de développement dans des équipes qui doivent être coordonnées.

Elle est adaptée à des projets de petite taille

#### **Crystal,**

Plus qu'une méthodologie, Crystal est un ensemble de méthodes peu contraignantes et adaptables à la variabilité de l'environnement de développement et aux caractéristiques spécifiques des projets.



Le risque principal lors de l'adoption de Crystal est la capacité des intervenants à déterminer comment appliquer les différentes méthodes proposées.

### **DSDM (Dynamic Systems Development Method)**

Un peu comme Crystal, DDSM est une approche de développement rapide d'une application qu'une méthode particulière. Surtout connue en Europe et en Grande Bretagne, DDSM possède une importante structure de support en place, de nombreux livres et sessions de cours lui sont consacrés.

Le processus DDSM possède 5 phases :

- ↳ Etude de faisabilité ;
- ↳ Etude métier ;
- ↳ Modèle fonctionnel (construit itérativement) ;
- ↳ Design et construction ;
- ↳ Implémentation.

Cette approche Agile définit les rôles et responsabilités de chacun mais se veut extensible et adaptatif selon le projet envisagé. Elle ressemble plus aux processus traditionnels de développement et de ce fait est plus facilement adoptable pour les développement basé sur les processus d'organisation.

### **XP (eXtreme Programming)**

Extreme Programming est une méthodologie destinée à des équipes de taille réduite qui développent des logiciels dans un contexte où les besoins sont mal connus ou susceptibles d'évoluer rapidement.

XP se démarque des approches traditionnelles en mettant l'accent sur l'implication du client, le travail d'équipe, l'adaptation au changement et la robustesse des applications développées.

La méthodologie XP s'articule autour de quatre valeurs fondamentales:

- ↳ Communication
  - priorité est donnée à la collaboration avec le client plutôt qu'aux négociations contractuelles.
  - client placé au centre du projet. Les développeurs communiquent directement avec le client plutôt que d'essayer d'interpréter des spécifications.
- ↳ Simplicité

XP fait le pari qu'il vaut mieux faire quelque chose de simple aujourd'hui et payer un peu plus demain pour l'adapter si nécessaire, que de faire quelque chose de plus compliqué aujourd'hui dont on n'aura peut-être jamais l'usage dans le futur.
- ↳ Feed-back

Par son principe d'itérations courtes, XP favorise le feed-back du client et des utilisateurs. Le pilotage du projet se fait par petites corrections successives.
- ↳ Courage

Il est essentiel que les programmeurs XP aient du courage:

  - Courage de détourner le regard des choses qu'on aura à implémenter dans le futur, pour se concentrer sur ce qui est essentiel aujourd'hui.
  - Courage de mettre à la poubelle du code complexe ou difficile à maintenir

- Courage d'introduire des changements parfois importants dans un système mis en production dès les premières phases du projet

L'Extreme Programming est aujourd'hui le "fer de lance" des méthodes agiles mais ne peut s'appliquer qu'à de petites équipes d'au maximum 12 personnes et de ce fait à des projets relativement petits.

### **UP (Unified Process)**

Le processus unifié est un processus de développement logiciel itératif, centré sur l'architecture, piloté par des cas d'utilisation et orienté vers la diminution des risques.

C'est un patron de processus pouvant être adapté à une large classe de systèmes logiciels, à différents domaines d'application, à différents types d'entreprises, à différents niveaux de compétences et à différentes tailles de l'entreprise.

L'objectif d'un processus unifié est de maîtriser la complexité des projets informatiques en diminuant les risques.

UP répète un certain nombre de fois une série de cycles (itérations) qui s'articule autour de 4 phases

#### ↳ Analyse des besoins

L'analyse des besoins donne une vue du projet sous forme de produit fini.

Cette phase porte essentiellement sur les besoins principaux (du point de vue de l'utilisateur) l'architecture générale du système, les risques majeurs, les délais et les coûts. On met en place le projet.

#### ↳ Elaboration

L'élaboration reprend les éléments de la phase d'analyse des besoins et les précise pour arriver à une spécification détaillée de la solution à mettre en œuvre.

L'élaboration permet de préciser la plupart des cas d'utilisation, de concevoir l'architecture du système et surtout de déterminer l'architecture de référence.

#### ↳ Construction

La construction est le moment où l'on construit le produit. L'architecture de référence se métamorphose en produit complet.

Le produit contient tous les cas d'utilisation que les chefs de projet, en accord avec les utilisateurs, ont décidé de mettre au point pour cette version.

#### ↳ Transition

Le produit est en version bêta. Un groupe d'utilisateurs essaye le produit et détecte les anomalies et défauts.

Cette phase suppose des activités comme la formation des utilisateurs clients, la mise en œuvre d'un service d'assistance et la correction des anomalies constatées.



Chaque phase est décomposée en activités qui participent à l'élaboration non seulement de la phase mais du projet dans son entièreté. Ces activités sont :

↳ Expression des besoins

L'expression des besoins comme son nom l'indique, permet de définir les différents besoins et fournir une liste de leurs fonctions. Ces besoins sont autant fonctionnels (du point de vue de l'utilisateur) que non fonctionnels (technique).

Le modèle de cas d'utilisation présente le système du point de vue de l'utilisateur et représente sous forme de cas d'utilisation et d'acteur, les besoins du client.

↳ Analyse

L'objectif de l'analyse est d'accéder à une compréhension des besoins et des exigences du client. Un modèle d'analyse livre une spécification complète des besoins issus des cas d'utilisation et les structure sous une forme qui facilite la compréhension (scénarios), la préparation (définition de l'architecture), la modification et la maintenance du futur système.

↳ Conception

La conception permet d'acquérir une compréhension approfondie des contraintes liées au langage de programmation, à l'utilisation des composants et au système d'exploitation.

Elle détermine les principales interfaces et les transcrit à l'aide d'une notation commune.

↳ Implémentation

L'implémentation est le résultat de la conception pour implémenter le système sous forme de composants, c'est-à-dire, de code source, de scripts, de binaires, d'exécutables et d'autres éléments du même type.

Les objectifs principaux de l'implémentation sont de planifier les intégrations des composants pour chaque itération, et de produire les classes et les sous-systèmes sous forme de codes sources.

↳ Test

Les tests permettent de vérifier des résultats de l'implémentation en testant la construction. Pour mener à bien ces tests, il faut les planifier pour chaque itération, les implémenter en créant des cas de tests, effectuer ces tests et prendre en compte le résultat de chacun.

UP est souvent considéré comme un guide peu flexible et à processus lourd et contraignant. Pourtant la plupart des caractéristiques Agiles sont incorporée dans la philosophie UP, mais le détail des processus a tendance à obscurcir ces derniers.

Un reproche souvent exprimé vis-à-vis d'UP (et de RUP, sa version commerciale d'IBM) est la lourdeur de la démarche lors de petit projet. C'est malheureusement le cas lorsque l'on tente d'implémenter UP tel quel sans en comprendre la souplesse et l'adaptabilité.

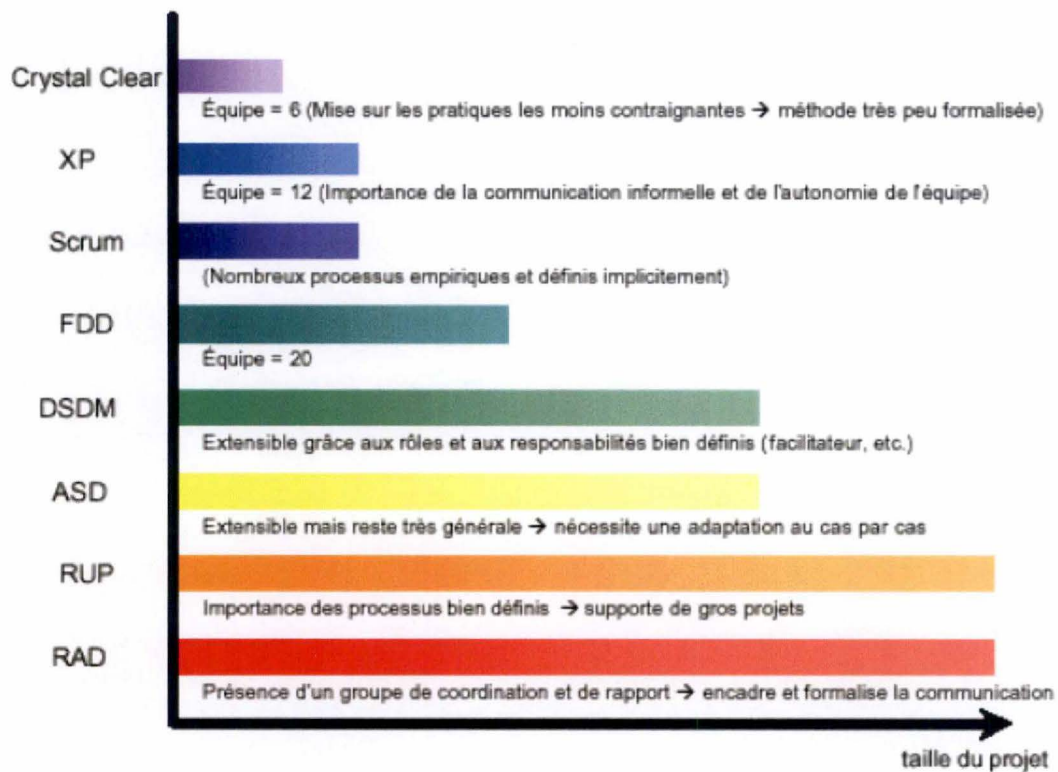


Figure 3-10 : Adéquation des méthodes agiles en fonction de la taille du projet

Compte tenu de l'importance en taille des projets mis en œuvre au S.P.F. Finances et de la volonté de coordination et de standardisation les méthodes RUP et RAD sont les deux méthodologies qui peuvent répondre aux besoins.



## • Définition des chapitres d'évaluation

<b>Critères d'évaluation</b>	<b>Poids</b>
A. Prix	35
B. Qualité de la solution logicielle	30
Fonctionnalités de gestion des spécifications	4
Analyse Conceptuelle	6
Fonctionnalités « Entreprise Data Modeling »	5
Fonctionnalités Design – IDE- Java 2EE	6
Fonctionnalités « Outils de Test »	4
Intégration des différents composants logiciels de l'offre, reprise de l'existant et environnement multiutilisateurs	5
C. Qualité de la méthodologie	15
Complétude méthodologie et complétude livrables proposés	8
Adaptation environnement S.P.F.	7
D. Intégration Méthode – Outils	5
E. Lisibilité et conformité de l'offre (5 %)	5
F. Case Study (5 %)	5
G. Formation – Documentation (3 %)	3
H.SLA – Support (2 %)	2
<b>TOTAL</b>	<b>100</b>

## • Définition des critères de choix de la méthodologie de référence

La méthodologie doit couvrir tous les cycles de développement d'une application.

Vu la complexité croissante de ces applications, la formalisation nécessaire est de plus en plus complexe, ainsi la méthodologie devra :

- ☞ Introduire d'un « langage commun » entre les différents intervenants
- ☞ La définition d'un ensemble de règles formelles, de diagrammes et de conventions auxquelles les livrables du projet devront répondre et ceci au cours des différentes étapes du développement. Ces règles formelles devront permettre de traduire et de formaliser les informations fournies par les intervenants « métier » en informations permettant la construction d'applications ICT.

Cette complexité rend la formalisation quasi impossible à réaliser sans le support d'outils spécialisés permettant au minimum:

- ☞ La réalisation de diagrammes complexes consistants et corrects
- ☞ Le partage et la consolidation de ceux-ci dans un projet et au travers des différents projets de l'entreprise
- ☞ La construction d'un dictionnaire de données commun à l'entreprise.
- ☞ Un support à la transformation des diagrammes en code exécutable

### 3.10.2. Evaluation des différentes méthodes candidates possibles

Malheureusement cette partie du projet c'est révélée réduite à sa plus simple expression étant donné le système d'appel d'offre en cours au S.P.F. Finances.

Nous avons reçu quatre offres en réponse au cahier des charges. L'évaluation des offres en fonction des critères de sélections cités ci-dessus se trouve en annexe.

Les quatre soumissionnaires proposaient comme choix méthodologique le processus unifié (UP) dans sa version libre ou dans sa version commerciale (RUP).

Le choix méthodologique effectué par les soumissionnaires répondait aux besoins et contraintes exprimés précédemment.

Pour rappel, les principaux besoins étaient :

- ☞ réaliser l'intégration interne et externe des systèmes d'information
- ☞ permettre de formaliser et de stabiliser le déroulement des projets dans un cadre commun en uniformisant :
  - le contenu des étapes de ceux-ci
  - les rôles des différents acteurs
  - les livraisons de chaque étape.
- ☞ introduire un langage commun entre les différents intervenants
- ☞ définir un ensemble de règles formelles, de diagrammes et de conventions auxquels les documentations et livraisons du projet devront répondre et ceci au cours des différentes étapes du développement.
- ☞ réaliser des diagrammes complexes consistants et corrects
- ☞ partager et consolider les modélisations dans un projet et à travers l'ensemble des différents projets de l'entreprise
- ☞ construire un dictionnaire de données commun à l'entreprise et un glossaire commun.
- ☞ supporter à la transformation des diagrammes en code exécutable.
- ☞ assurer la gestion des spécifications
- ☞ permettre l'analyse conceptuelle
- ☞ favoriser le design technique
- ☞ modéliser les données.
- ☞ assurer la programmation en respectant les standards
- ☞ construire et réaliser les tests
- ☞ assurer le déploiement
- ☞ mettre en place un environnement de travail collaboratif, une gestion des versions ...



Quant aux contraintes, elles s'orientaient autour de cinq axes :

- ✧ l'organisation du S.P.F. Finances ;
- ✧ la prise en compte de la mentalité du personnel du S.P.F. Finances ;
- ✧ la restructuration organisationnelle et fonctionnelle du S.P.F. ;
- ✧ le fonctionnement par appel à des sociétés externes ;
- ✧ l'intégration de l'approche Métier et ICT.

### **3.10.3. Choix de la méthode**

La méthodologie proposée par le soumissionnaire lauréat est UP en tenant compte des pratiques développées dans sa version commerciale RUP.

La méthodologie adaptées aux spécificités du S.P.F. Finances sera baptisée FUP pour « Finances Unified Process ».

### **3.10.4. Processus d'adaptation de la méthode pour répondre aux besoins du S.P.F. Finances.**

- 1- Assessment Méthodologie (préétude interne)

#### **Méthode et buts de l'Assessment :**

L'Assessment a pour but de rassembler les informations à propos des méthodes de développement utilisées jusqu'à présent au sein du S.P.F. Finance ; il capte ainsi les diverses informations intéressantes relatives à la méthodologie appliquée jusqu'à présent et pouvant impacter la nouvelle méthodologie.

L'Assessment a été réalisé sur base de documentations reçues et de diverses interviews réalisées auprès des personnes responsables des divers développements informatiques au sein du S.P.F. Finances. Ensuite celles-ci ont été analysées, structurées et rassemblées en divers points qui sont le résultat de l'Assessment.

Cette étude préalable a tenté de collecter les informations suivantes :

A. Liste des projets : Les projets en cours de développement ou en "maintenance" et les nouveaux projets à venir, soit les informations suivantes :

- ✧ Nom, date de début, date de fin
- ✧ Nombre de ressources participantes (=nombre moyen de participants sur le projet)
- ✧ Type de projet (=début de catégorisation de projet) ou éventuellement une petite description
- ✧ Sponsor (Sponsor IT, Métier ou autres)
- ✧ Technologie utilisée pour la livraison, le développement ou l'intégration
- ✧ Méthodologie utilisée (brève description)

B. Liste des ressources : Pour nos besoins, nous ne devons pas avoir une liste exhaustive des ressources humaines IT mais nous voulions connaître:

↵ Le nombre de ressources pour un type de discipline. Il a été jugé bon de reprendre les disciplines/travailleurs proposée par RUP, à savoir les principaux:

- Chef de Projet,
- Analyste système,
- Spécificateur de use cases,
- Concepteur interface utilisateur,
- Analyste métier,
- Architecte,
- Designer,
- Développeur,
- Intégrateur,
- Concepteur de test,
- Testeur,
- Concepteur de base de données,
- Rédacteur technique,
- Développeur de cours,
- Responsable de la configuration-Administrateur d'outil,
- Responsable du déploiement.

On devra sans doute rassembler certains travailleurs ou créer de nouvelles catégories afin d'être plus proches des réalités du S.P.F.

↵ Le niveau de compétence dans ce type de discipline. Le niveau de compétence pourrait être exprimé par une liste de formations suivies, un nombre d'années d'expérience dans un domaine ou avec un outil, ou encore des certifications acquises.

C. Liste des outils actuels, à savoir :

- ↵ Nom
- ↵ Catégorie (=il s'agit du rôle de l'outil. Ex: développement, test, gestion source, etc.)
- ↵ Nombre d'utilisateurs (=il s'agit d'avoir une idée de l'usage de l'outil)
- ↵ Compétence ou niveau d'intégration de l'outil au S.P.F. (= il s'agit d'avoir une idée de la compétence à utiliser l'outil)

D. Liste des méthodologies actuelles, soit le nom et la description succincte de celle-ci.

E. Outils actuels de développement



### Conclusion de l'Assessment

- ✧ Tenir compte des résultats du BPR tant au niveau organisationnel que fonctionnel.
- ✧ Assurer la correspondance rôles COPERFIN et travailleur/rôles de la méthodologie.
- ✧ Tenir compte des projets actuellement en développement.
- ✧ Assurer la formation des analystes
- ✧ Définir « Quantité » et niveau de détail des livrables à fournir dans un projet
- ✧ Assurer la maintenance des produits/solutions et prévoir une méthodologie spécifique (la maintenance ne fait pas partie de la méthodologie actuellement en cours d'élaboration)
- ✧ Imposer au fournisseur la méthodologie interne du S.P.F. Finances
- ✧ Différencier : Méthodologie de développement  $\diamond$  Gestion de projet
- ✧ Il existe actuellement un système de gestion du changement qui semble correspondre à une gestion de programme (de projet). Est-ce la bonne approche par rapport à une méthodologie ?
- ✧ Définir les jalons/critères de fin de phase
- ✧ Définir la longueur des itérations ?
- ✧ Définir les phases de développement et/ou en ajouter
- ✧ Tenir compte que les participants à la phase de préétude ne sont pas toujours les mêmes que ceux de la phase de développement
- ✧ Définir le contenu du cahier des charges
- ✧ Réutiliser les livrables de la phase de préétude
- ✧ Organiser Préétude et développement dans la méthodologie
- ✧ Tenir compte de l'approche due à l'appel d'offre
- ✧ Mettre sur pied d'une procédure de communication et de gestion du changement impliquée par la méthodologie, étant donné qu'elle devrait impliquer à terme tous les départements ICT, les départements d'analyse et les participants externes.
- ✧ Quid de la méthodologie pour les projets « legacy » ?

- 2- Prise en compte des profils métiers (orientation métier)

### Profils existants

- ✍ **Chefs de projets** : A l'ICT du S.P.F. Finance, le rôle de chef de projets est reconnu. Il n'est cependant pas certain que les chefs de projets sont tous au courant des concepts importants liés aux nouvelles technologies (approches orientées objets, développement itératifs, etc.). Par contre il semble que les chefs de projets (ICT et métiers) comprennent l'approche « use case ».
- ✍ **Développeurs** : Sur l'ensemble des ressources faisant partie du développement à l'ICT, environ 200 personnes sont issues du monde mainframe (analystes et/ou développeurs), et seulement une très petite cinquantaine de personnes sont probablement formées ou aptes à travailler avec les nouvelles technologies et les pratiques en relation avec ces nouvelles technologies (approches orientées objets, développement itératifs, etc.). Il y a une espérance de former ou de transformer environ 30% des développeurs actuels dans les nouvelles technologies.
- ✍ **Analystes fonctionnels** : Les profils d'analystes fonctionnels sont actuellement soit faisant partie de l'ICT (Développement) soit de l'administration. Il semble que les analystes fonctionnels « patrimoine » font actuellement partie de l'ICT alors que les analystes fonctionnels « taxation et recouvrement » font encore partie de l'administration.
- ✍ **Les disciplines relatives aux tests** sont les disciplines les plus « faibles » au S.P.F. Finances, dans le sens où le rôle n'est pas reconnu comme tel et que peu de méthodologie, de rigueur ou de structuration est mise en œuvre.

### CoperFin a défini des rôles

COPERFIN a défini un ensemble de rôles que l'ICT devrait appliquer ou supporter.

Rôles définis par CoperFin	Groupes
Chef de projet senior	Chefs de projet
Chef de projet	
Chef de projet junior	
Analyste d'application	Analyste technique
Designer d'application	
Designer d'infrastructure	
Senior Programmeur	Programmeurs
Programmeur	
Senior Programmeur Infrastructure	
Testeur	Testeur



Rôles définis par CoperFin		Groupes
Responsable système	Responsable technique opérationnel	
Responsable système senior		
Expert technique		
Inspecteur senior	Intégrateur	
Agent administratif senior	Analyste	
Agent administratif		
Correspondant externe		
Analyste	Analyste métier	

Les rôles ainsi définis par CoperFin regroupent souvent plusieurs profils existant dans l'organisation. Une des premières missions de SupDev sera de mettre en place non seulement les nouveaux rôles dans le cadre méthodologique choisi, mais de s'assurer de la formation et de la compétence des personnes assignées à ces rôles.

- 3- Création d'un groupe de travail et de réflexion méthodologique (« Core Team »)

#### **Formation du « Core Team »**

Le « Core Team » est le noyau central du projet SupDev. C'est un groupe de travail et de réflexion méthodologique dont la mission principale est de mettre au point la méthode adaptée au SFP Finances.

Ce groupe de travail est composé de cinq membres interne au S.P.F. et de trois consultants de la firme soumissionnaire.

Les membres du S.P.F. sont choisis comme suit :

- ↗ Un membre des services d'analyse métier ;
- ↗ Un membre des services IT, section Infrastructure ;
- ↗ Un membre des services IT, section Exploitation ;
- ↗ Deux membres des services IT, section Développement.

Les membres de la société soumissionnaire sont choisis comme suit :

- ↗ Un membre est spécialiste méthodologique ;
- ↗ Un membre est spécialisé dans les outils de développement choisis ;
- ↗ Un membre est un spécialiste de l'implémentation de méthodologie dans des environnements gouvernementaux.

Ce « Core Team » peut à tout moment faire appel à des ressources tant internes qu'externes lorsqu'il en trouvera l'utilité.

#### **Méthode de travail du « Core Team »**

La méthode de travail de ce groupe sera la suivante :

- ↗ Formation à la méthodologie retenue (UP) et à sa version commerciale RUP. (2 semaines à raison de 3 jours par semaine)
- ↗ Réalisation de l'Assessment (2 semaines à temps plein) ;
- ↗ Conclusion de l'Assessment et reportage au comité de Pilotage et au comité de Direction ;
- ↗ Go/No Go du comité de pilotage ;
- ↗ Développement itératif de la méthodologie adaptée au S.P.F. Finances (6 mois)
- ↗ définitions et analyses détaillées des différentes étapes de développement ;
- ↗ adaptation des rôles et fonctions en adéquation avec les principes de CoperFin ;



- ↳ réalisation des modes opératoires ;
- ↳ Démarrage des projets pilotes ;
- ↳ Révision méthodologique si nécessaire ;
- ↳ Mise en application des nouveaux standards méthodologiques.
- ↳

#### • 4- Test de la méthodologie au travers de projets pilotes

Pour tous les projets développés au S.P.F. Finances, la mise en production du produit final est toujours précédée d'une phase « pilote » qui permet de le tester dans un environnement de production réel.

En général, il s'agit de tester le produit final à une échelle réduite (nombre d'utilisateurs restreint, confinement géographique) afin de déterminer avec justesse l'adéquation des développements avec les besoins « métier » et les contraintes techniques y afférentes.

Dans le cadre de la mise en place d'un standard méthodologique, il s'agira de tester la méthode sur un projet réel à développer. Cependant, il apparaît très vite que beaucoup de projets orientés nouvelle technologie devront aussi s'adapter à la nouvelle méthodologie et aux nouveaux outils de développement. C'est pourquoi il sera aussi procédé au test de la méthodologie à travers l'implémentation de celle-ci dans un projet déjà en cours de développement.

### **3.11. Rôles et responsabilités des parties prenantes**

#### • Rôles et responsabilités du S.P.F. Finances

Le S.P.F. Finances s'engage à fournir au projet SupDev toutes les ressources nécessaires au bon accomplissement de sa mission. Cela comprend non seulement les ressources matérielles, mais aussi les ressources humaines et budgétaires.

Il est responsable de l'analyse des besoins et des spécificités. Cette analyse sera menée par le Core Team aidé en cela par les consultants de la société soumissionnaire qui auront la tâche de le guider et de le conseiller.

Une fois la méthodologie créée, le S.P.F. Finances aura la responsabilité de la mettre en application dans ses structures et de veiller à l'adaptation organisationnelle et fonctionnelle nécessaires.

#### • Rôles et responsabilités du soumissionnaire

Le soumissionnaire s'engage à fournir au S.P.F. Finances son expertise méthodologique et toutes les ressources nécessaires aux transferts de connaissance et de compétence vis-à-vis du Core Team pour l'aider dans sa mission.

Il est également responsable de la production des livrables, de l'implémentation technique des outils de modélisation (logicielle uniquement, le S.P.F. Finances prenant à sa charge l'environnement matériel) et de leur maintenance. Il est également tenu de fournir de l'assistance technique aux utilisateurs de la suite logicielle.

### 3.12. Budget

Pour des raisons de confidentialité, le budget ne sera pas mentionné.

La manière de fonctionner au S.P.F. Finances est de travailler par budget fixe déterminé en fonction de l'attribution du marché. En accord avec les exigences du cahier des charges, les soumissionnaires remettent offre et fixent ainsi le montant total du budget du projet selon leur vision de celui-ci et leur politique de prix.

Lors de l'attribution du marché, la dimension « prix » revêt une importance de 35% sur l'évaluation totale de l'offre.

Une fois le marché attribué par la commission d'évaluation du S.P.F. Finances, un rapport d'attribution est adressé à l'Inspection des Finances qui remet un avis sur l'attribution et sur les budgets en jeu. Ensuite, le dossier est présenté au Conseil des Ministres, qui l'avalise ou non. En cas d'avis favorable du Conseil des Ministres, le marché est officiellement attribué et le budget est fixé une fois pour toute.

S'il s'avérait durant le déroulement du projet qu'une extension de budget fût nécessaire pour accomplir des tâches non prévues explicitement dans le cahier des charges, il faut demander un « change request » (demande de changement, d'extension) au comité de pilotage du projet. Ce « change request » doit être valorisé budgétairement parlant et la procédure d'acceptation est la même que pour celle de l'attribution d'un marché (Inspection des Finances et Conseil des Ministres).

Dans le cadre du projet SupDev, vous trouverez ci-après les critères d'évaluation budgétaire de l'offre.

#### ↳ Suite logicielle

##### Logiciels et extensions éventuelles

Dénomination du produit + version :

OS supportés :

Description du produit et besoin auquel il répond (réf. au CSC) :

Remarques relatives à la licence :

Prix de la licence : tableau ci-dessous (détailler par année s'il y a lieu) :

Prix de la maintenance (mises-à-jour, patch, upgrade, major upgrade, releases, ...) : tableau ci-dessous (détailler par année s'il y a lieu)

Non couvert par la licence / maintenance :

Nom du logiciel + version	Quantité	Prix unitaire HTVA	Montant total licences HTVA	Prix unitaire Maintenance annuelle HTVA	Montant total maintenance annuelle HTVA	Taux de TVA
	50 (tranche fixe)					
	25 (incrément)					
	50 (incrément)					

#### ↳ Prestation de service

SERVICES (1)	Quantité	Prix / homme jour HTVA	Profils (2)	Total HTVA	Taux de TVA
Installation – Paramétrisation	190 H/J				
Définition – Implémentation Méthodologie					
Coaching	250 H/J				
Incrément Coaching	25 H/J				
Autres					





## Formations aux outils

DENOMINATION	Prix par jour par participant HTVA	Durée	Montant Total HTVA	Réduction sur quantité	Taux de TVA
Formation Analyste d'application					
Formation Application Designer					
Formation Programmeur Senior					
Formation Programmeur					
Formation Testeur					
Formation DBA					
Formation Administrateur					
Formation « Cellule Formation et Support »					



## Formations à la méthodologie

DENOMINATION	Prix par jour par participant HTVA	Durée	Montant Total HTVA	Réduction sur quantité	Taux de TVA
Formation Analyste d'application					
Formation Application Designer					
Formation Programmeur Senior					
Formation Programmeur					
Formation Testeur					
Formation DBA					
Formation Administrateur					
Formation « Cellule Formation et Support »					

### **3.13. Structure d'accompagnement et d'encadrement**

#### **3.13.1. Propriétaire du projet**

Le propriétaire du projet est le S.P.F. Finances.

Les études, architectures et développements éventuellement développés par le personnel de l'adjudicataire et la documentation qui les accompagne, et, d'une manière générale, tout document établi directement ou indirectement en exécution du présent marché par le personnel de l'adjudicataire, ainsi que les droits de propriété intellectuelle qui y sont associés, sont, au fur et à mesure de leur établissement, la propriété du S.P.F. Finances.

Il est interdit au personnel de l'adjudicataire d'emporter avec soi tout document appartenant au S.P.F. Finances, sauf dans les cas où cela serait indispensable à l'exécution de sa mission. Ce serait notamment le cas lors des déplacements entre différents sites du S.P.F. Finances.

**Figure 3-11 : Disposition légales en matière de propriété**

#### **3.13.2. Le comité de Direction**

Le Comité de Direction est l'instance décisionnelle suprême de l'organisation. Il est composé du Président du S.P.F., des manager N1 et des directeurs généraux.

Outre ses missions de gestion du département Finances, ce comité a également des attributions décisionnelles dans le cadre de la gestion des projets. En effet le comité de Direction est le siège de toutes les décisions stratégiques et politiques du département.

Grâce au service PMO (program manager office) et aux managers de projet (il y en a un par entité N-1) ce comité est mis au courant de l'évolution des projets et des différents axes stratégiques développés.

Ce comité se réunit une fois par mois.

#### **3.13.3. Le comité de Pilotage (Steering-committee)**

Le comité de pilotage comprend, d'une part, le responsable du projet chez l'adjudicataire, accompagné des différents chefs de développement concernés selon l'état d'avancement du projet, et, d'autre part, la Cellule d'accompagnement du projet représentant l'Administration.

La Cellule d'accompagnement est la seule instance à laquelle l'adjudicataire doit rendre des comptes ; elle se fait assister par le groupe de projet interne qui est chargé du suivi de l'exécution du marché.

Le comité de pilotage a un pouvoir très large : il traite les aspects contractuels, administratifs et techniques de l'exécution du marché (livraison, réceptions, garantie, avancement du projet, qualité des exécutions...). Il est autorisé à accepter des modifications de l'offre initiale sans devoir dépasser le prix initial.



Il est le siège du reporting et périodiquement l'adjudicataire fait un rapport complet au comité de pilotage (steering-committee) de l'avancement du projet (état de chaque étape de réalisation), des problèmes rencontrés avec propositions de solution.

Le reporting est rédigé par le personnel de l'adjudicataire. Il est soumis à l'approbation préalable du chef de projet du S.P.F. Finances qui se chargera de sa diffusion aux membres du steering par messagerie électronique.

Ce comité se réunit une fois par mois mais peut être convoqué à tout moment si le besoin s'en fait sentir.

## 4. Etat d'avancement du développement de la méthodologie

### 4.1. Aspects Méthode

#### 4.1.1. Le cycle de vie

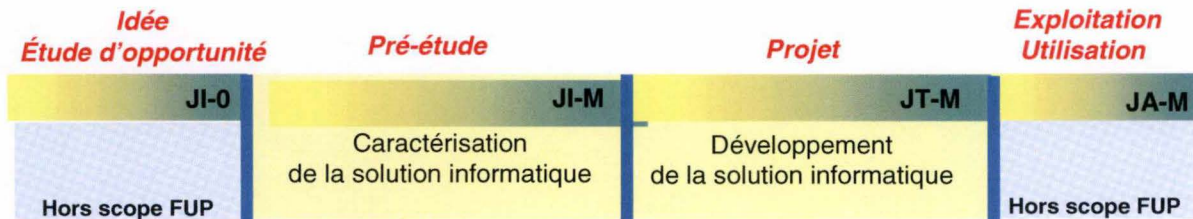


Figure 4-1 : Le cycle de vie version S.P.F. Finances

Avant d'être en développement, un projet au S.P.F. Finances est imaginé, étudié et initialisé lorsque l'opportunité du projet a été étudiée. Parfois il est proposé à un marché public. Un projet existe donc pour toute création d'un nouveau système d'information mais également pour un ensemble de modifications (versions) d'un système d'information existant (maintenance/évolution du produit).

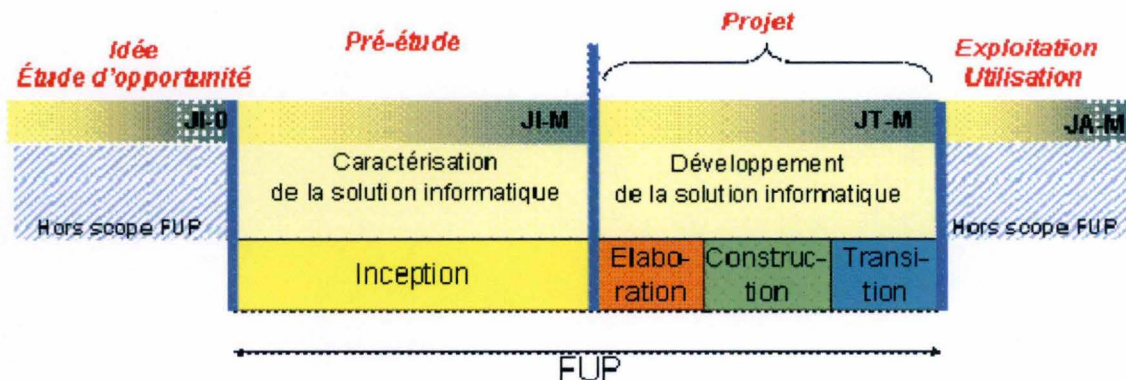


Figure 4-2 : Le cycle de vie de FUP inspiré de RUP

Le cycle de vie du projet doit donc tenir compte de ces aspects : les phases UP doivent s'intégrer dans un cycle de vie plus large que le simple développement d'une solution informatique. Le cycle de vie proposé par UP est donc adapté au S.P.F. Finances.

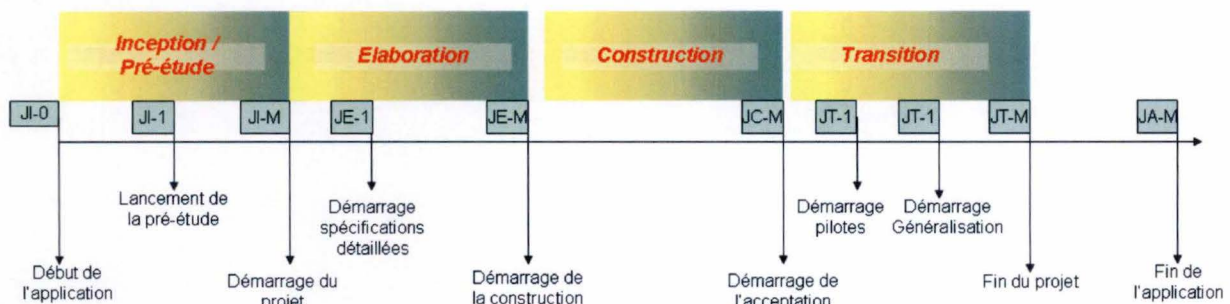


Figure 4-3 : Les principaux jalons du cycle de vie



En simplifiant l'approche, cela donne les phases et l'expression des jalons suivants :

	Projet S.P.F. Finances	Phases de FUP	Jalons
Le cycle de vie est initié.			JI-0
Le cycle de vie au S.P.F. Finances démarre d'abord par une préétude. Par rapport à UP, cette préétude couvre la phase d'inception. Dans le cadre du S.P.F. Finances, cela permettra la rédaction d'un cahier des charges de développement et la sélection d'un fournisseur (si ce projet est externe) pour réaliser le développement de l'application.	« Pré étude »	Inception	
Les objectifs du projet sont bien connus. Un cahier des charges a été délivré. Il inclut les objectifs et limites du projet, le métier est éventuellement décrit, une liste des uses cases est fournie et une description des use cases critiques est définie. La solution proposée propose ou confirme l'architecture candidate et définit les coûts et plan du projet.			JI-M
Le projet de développement prend place lors du début de la phase d'Elaboration et démarre par une itération qui permettra de « consolider » (rassembler ou encore fonder) l'ensemble des travaux réalisés en phase d'Inception (préétude). En effet, lorsque le projet est mis en œuvre par un fournisseur externe, il s'écoule une période de plusieurs mois entre la fin de la phase d'Inception et le début de la phase d'Elaboration.	Projet	Elaboration	JE-1
		<i>Itération de consolidation</i>	
L'environnement de travail est prêt et est accessible aux membres du projet. Il contient tous les artefacts nécessaires au démarrage du projet et ils sont assimilés par les participants.			
La phase d'Elaboration va ensuite être mise à profit pour construire une première version de l'application sur un périmètre fonctionnel limité. En fin de phase d'Elaboration une version prototype de l'application doit être produite et validée, confirmant également la faisabilité de l'architecture.		<i>Itérations d'élaboration</i>	JE-x
L'architecture du système est stable : des tests ont montré sur un « prototype » que les principaux risques ont été adressés. Une grande partie des uses cases sont documentés.			JE-M
Ensuite la phase de Construction finalisera l'ensemble des développements de l'application et amènera l'application au niveau de qualité requis. Pendant cette phase des points de contrôle formels à chaque itération peuvent être programmés pour permettre de valider les opérations de développement et la qualité de ce qui est développé, testé et intégré		Itérations de construction	JC-x
La totalité des uses cases est décrite et le système est livré en release « Beta », ce qui inclut l'analyse, le design, l'implémentation et les tests d'intégration de l'application.			JC-M
La dernière phase, Transition, permettra la réception (validation) de l'application, la mise en production du (ou des) site(s) pilote(s) et le déploiement général.		Itérations de transition	
Tests de réception			JT-1
Fin des pilotes			JT-2
Généralisation du déploiement			JT-M

Il est important de rappeler que ce processus est itératif, et que la majorité des disciplines sont donc mise en œuvre à chaque itération.

## La maintenance d'une application informatique

Lorsqu'un système informatique est en production, il devra probablement évoluer et des changements sont à prendre en compte. Ces changements ou transformations sont gérés par le processus transverse Exploitation / Maintenance /Support.

Ce processus décide du rythme des évolutions de l'application en fonction des besoins et dès qu'un ensemble d'évolutions ou de changements sont acceptés et décidés, un nouveau projet prend place et une nouvelle instance de FUP est initiée.

Il est donc important de comprendre que le processus de maintenance d'une application suit exactement la méthodologie FUP.

Les disciplines et processus suivants apparaissent dans le cycle de vie d'un projet au S.P.F. Finances.

### 4.1.2. Les Disciplines

Afin d'aider les participants au projet à comprendre l'approche proposée dans chaque discipline, un document « discipline » est proposé pour chacune d'elle. Il décrit les activités à exécuter et les artefacts à fournir par la discipline.

Discipline/Processus FUP	Participants principaux	Document « Discipline »	Commentaires
Instruction du projet	Chef de projet métier Chef de projet IT	/	Processus hors méthodologie FUP et UP/RUP.
Modélisation métier	Chef de projet Métier Analyste fonctionnel	<b>FUP-Discipline Modélisation Métier</b>	
Expression des besoins / Exigences	Chef de projet Métier Analyste fonctionnel	<b>FUP-Discipline Exigences</b>	
Analyse / Conception	Architecte Officier Sécurité Analyste technique	<b>FUP-Discipline Analyse &amp; Conception</b>	
Implémentation	Programmeur développement	<b>FUP-Discipline Implémentation</b>	
Test	Testeur	<b>FUP-Discipline Test</b>	
Déploiement	Programmeur opérationnel	/	Dans cette version de la méthodologie, cette discipline n'est pas documentée. Il faudra se référer à UP
Gestion des configurations et changement	Responsable système Chef de projet ICT	/	Dans cette version de la méthodologie, cette discipline n'est pas documentée. Il faudra se référer à UP
Gestion des projets	Chef de projet ICT	/	Discipline hors méthodologie FUP. Elle sera détaillée dans la méthodologie de gestion de projet



Discipline/Processus FUP	Participants principaux	Document « Discipline »	Commentaires
Environnement	/	/	Discipline hors méthodologie FUP. Il faudra se référer à UP
Exploitation / Maintenance / Support	/	/	Processus hors méthodologie FUP et UP/RUP
Acquisition	/	/	Processus hors méthodologie FUP et UP/RUP

Si nous présentons ici les participants principaux à chaque discipline, il faut être conscient que pratiquement d'autres participants interviennent probablement dans l'exécution de la discipline et il faut se référer aux documents disciplines pour en connaître les détails.

Il est également important de noter que la discipline de gestion des projets ne fera concrètement pas partie de cette méthodologie : elle sera prise en compte par la mise en œuvre d'une méthodologie de gestion de projet.

Nous constatons enfin que par rapport aux disciplines UP, de nouveaux processus sont proposés : Instruction du projet, Exploitation/Maintenance/Support et Acquisition.

- ↳ Instruction du projet : Ce processus a pour finalité d'initier, étudier puis proposer une ou plusieurs solutions au sein du système d'information existant. Cela se concrétisant normalement par une préétude et un projet FUP.
- ↳ Exploitation / Maintenance et Support : Ce processus définit les exigences en matière d'exploitation et de maintenance des applications développées et mises en production. Elle comprend en outre le soutien et l'information des utilisateurs (help desk, ...), la correction et/ou la modification des applications (voir le chapitre 0 La maintenance d'une application informatique) et la surveillance et le maintien de la qualité de service de l'application en production.
- ↳ Acquisition : Cette discipline a pour finalité de maîtriser les opérations faisant appel à des fournisseurs externes pour obtenir l'application et/ou les services satisfaisant à un besoin exprimé dans d'autres disciplines.

Ces disciplines sont normalement mises en œuvre par le S.P.F. Finances, mais étant donné qu'elles sont nettement hors du processus de développement, elles ne font pas partie de la méthodologie FUP.

### 4.1.3. Les modes opératoires

Chaque discipline ou processus transverse utilise un ou un ensemble des outils présentés ci-dessus.

Afin d'aider les participants au projet à délivrer les artéfacts avec les outils et selon l'approche proposée dans chaque discipline ou processus transverse, un mode opératoire (document) est proposé pour chacun de ceux-ci.

Discipline ou processus FUP	Outils principaux	Document « Mode Opératoire »
<i>Instruction du projet</i>	/	/
Modélisation métier	Borland CaliberRM Borland Together Control Center	MO-Discipline modélisation métier
Expression des besoins / Exigences	Borland CaliberRM Borland Together Control Center	MO-Discipline exigences.
Analyse / Conception	JBUILDER Studio E/R Studio	MO-Discipline analyse
Implémentation	JBUILDER Studio E/R Studio StartTeam	Cette version de FUP ne contient pas encore ce document.
Tests	Mercury Quality Center (comprend Test Director et Quicktest Professional)	Cette version de FUP ne contient pas encore ce document.
Déploiement	StartTeam	Cette version de FUP ne contient pas encore ce document.
Gestion des configurations et changement	Borland CaliberRM StartTeam	Cette version de FUP ne contient pas encore ce document.
Gestion des projets	/	/
Environnement	/	/
Exploitation / Maintenance / Support	/	/
Acquisition	/	/



#### 4.1.4. Les concepts transversaux

- Le glossaire des concepts métiers

##### A. Objectifs :

- ↳ Disposer d'un répertoire centralisé des concepts métiers particuliers et communs ;
- ↳ Mettre à disposition du S.P.F. un outil de contrôle des concepts métiers qui permettra d'assurer le suivi des changements actuels et futurs en permettant une analyse d'impact de premier niveau ;
- ↳ Assurer l'homogénéité dans la compréhension et la manipulation des concepts métiers ;
- ↳ S'assurer de la couverture totale des besoins métiers et de la non redondance des traitements au travers des différentes applications et procédures de travail.

##### B. Informations visées :

Les informations qui seront contenues dans le glossaire concernent les différents concepts métiers soit actuellement en cours au sein du S.P.F. Finance, soit définis au cours de différentes études menées durant l'élaboration des nouveaux processus de travail à l'occasion des BPR.

- Les conventions de nommages

Le S.P.F. Finances est une entité fédérée où se côtoient du personnel issus des trois rôles linguistiques à savoir Français, Néerlandais et Allemand.

Afin d'assurer une compréhension commune des dénominations des termes modélisés, nous avons mis en chantier la création de conventions de nommage de ses termes sur base d'acronymes.

Ces acronymes, abréviation de 3 à 5 positions sont créés à partir d'une des langues officielles et s'impose pour tous l'organisation après validation par un comité de contrôle. (Pour plus de détail nous renvoyons le lecteur à l'annexe 12.3 « les convention de nommage »).

Les conventions ont été établies afin de rencontrer les objectifs suivants :

- ↳ Souplesse d'utilisation ;
- ↳ Facilitation du suivi et de la maintenance des modélisations ;
- ↳ Lisibilité aisée après un temps d'adaptation et d'écologie ;
- ↳ Traçabilité complète des noms depuis la modélisation jusqu'à la programmation et les bases de données.

## • Le dictionnaire des données

### A. Objectifs :

- ↳ Disposer d'un répertoire centralisé des informations manipulées par les différents projets (classes, attributs, méthodes, tables, ....) ;
- ↳ Mettre à disposition du S.P.F. un outil de contrôle des redondances d'information et de suivi des conventions de nommage ;
- ↳ Pousser les analystes à utiliser des composants déjà existants ;
- ↳ Favoriser une approche homogène en ce qui concerne la définition et la structure des données (data type) ;
- ↳ Mettre à disposition du Métier et de l'IT un outil d'analyse d'impact de premier niveau ;
- ↳ Mettre à disposition un outil permettant de faciliter la maintenance des bases de données développées au S.P.F. Finances.

### B. Informations visées

Le dictionnaire des données se compose de deux parties étroitement liées mais destinées à deux types de publics différents.

La première partie se compose des éléments issus de l'analyse métier des différents concepts utilisés au sein du S.P.F. Finances. C'est l'approche métier concrétisée par les classes, leurs attributs et leurs méthodes.

La seconde partie se compose des éléments issus de la conception technique de l'approche métier au travers des bases de données qui en découlent. Cette approche se traduit par la conception du schéma entités-relations, avec ses tables et ses champs et ses règles de maintenance.

Ces deux parties sont étroitement liées et interdépendantes. Ainsi, l'utilisation de certains concepts métiers au travers des classes par l'appel à leurs méthodes de travail entraîne automatiquement l'accès aux données sous-jacentes à ces concepts.

Les règles associées aux diverses manipulations des données sont extrêmement importantes eu égard à la complexité des différentes règles métiers. Entre autres choses, les concepts de source authentique, de règles de protection de la vie privée, d'autorisation de modification, de cohérence des données, en sont quelques exemples parlants.

La mise en place de ce dictionnaire des données et l'obligation d'y recourir pour tout nouveau développement informatique permettront d'assurer au S.P.F. Finances l'homogénéité des traitements métiers de ces données.



## 4.2. Avancement des Projets pilotes

### 4.2.1. Critères de sélection

Initialement nous avions prévu la mise en place d'un projet pilote couvrant à la fois des aspects « métiers » demandant une analyse détaillée au travers de Use Cases et couvrant également les concepts IT standards du S.P.F. Finances.

Ce projet devait donc répondre aux critères suivants :

- ↳ Les Cas d'Utilisation sont au centre de la démarche
  - La gestion des exigences est réalisée dans un référentiel d'exigences
  - La traçabilité des exigences est réalisée vis-à-vis des cas de tests
- ↳ Une approche bien comprise de l'itératif et de l'incrémental
- ↳ Les standards du S.P.F. Finances sont respectés

Cependant, au cours de l'établissement des critères de sélection du projet « pilote », il est apparu que pour que celui-ci soit gérable et réalisable dans un délai de développement acceptable, il devait être de taille assez réduite. Cette constatation somme toute assez banale souleva alors un débat assez pointu sur la nécessité de tester la méthodologie FUP sur des projets de plus grande ampleur.

Mais les risques liés à l'implémentation d'une méthodologie en cours d'affinement dans un projet d'investissement de plusieurs millions d'euro étaient énormes.

Il s'avère qu'à l'époque, début 2005, un projet de grande ampleur était mis en place après avoir été sujet d'une préétude complète. Ce projet, dans son cahier des charges d'implémentation prévoyait déjà l'utilisation d'une suite logicielle de développement basée sur les outils Borland (en grande partie ceux retenues par le projet SupDev) et utilisant la méthodologie RUP. Ce projet prévoyait également en interne l'adaptation de la méthodologie aux besoins et attentes du S.P.F. Finances.

L'inconvénient majeur du choix de ce projet était qu'il avait démarré anticipativement à la standardisation méthodologique initiée par SupDev. Les efforts de rattrapage et d'adaptation demandés pour appliquer à la lettre les principes de FUP allaient avoir un impact non négligeable sur les plannings et l'utilisation des ressources de ce projet.

Le Comité de Pilotage de SupDev commanda alors une étude de faisabilité et d'impacts complète dont les conclusions furent présentées au Comité de Direction du S.P.F. Finances.

La décision fut alors prise de prendre également ce « gros » projet d'investissement comme projet « pilote » mais d'aligner sa méthodologie en fonction de son état d'avancement sans recourir systématiquement au rattrapage complet de ce qui était déjà réalisé.

Les seules adaptations imposées furent l'adoption des conventions standardisées de nommage, l'utilisation du glossaire métier et du dictionnaire des données transversaux.

#### **4.2.2. Projets retenus**

##### *La gestion centralisée des fiches cadastrales numérisées*

Le projet « gestion centralisée des fiches cadastrales numérisées » a pour but de mettre à disposition via l'intranet les fiches cadastrales numérisées pour tous les fonctionnaires travaillant pour le département Documentation Patrimoniale du S.P.F. Finances. Cette application sera aussi mise à disposition des communes et devra remplacer l'envoi d'information via Cdroms.

Ce projet sera développé entièrement avec les ressources propres du S.P.F. Finances.

Son développement (de l'analyse à l'implémentation) sera entièrement régit par la méthodologie FUP développée au sein de l'organisation.

Ce projet a comme ressources :

- un Chef de projet issu du métier
- un chef de projet ICT
- trois analystes
- un architecte fonctionnel et technique
- trois développeurs
- deux testeurs

La durée estimée du projet est de neuf mois à dater de novembre 2005.

##### *Système de Traitement intégré, Impôts et Recouvrement.*

Le projet « Traitement intégré Impôts et Recouvrement » concerne la réalisation d'un système de traitement intégré de toutes les transactions fiscales et non-fiscales en relation avec les citoyens et tous les autres contribuables dans le cadre de l'entité « Impôts et recouvrement ».

A la base du système se trouve le « Dossier fiscal unique », qui regroupe toutes les informations (fiscales et non-fiscales) du contribuable. A terme, le système de traitement intégré permettra au S.P.F. Finances d'assurer le traitement intégré de tous les principaux impôts.

La première phase du projet a été adjugée fin 2004, et la réalisation a commencé début 2005.

Le planning de ce projet est établi comme suit :

- février 2007 : mise en production du Dossier Fiscal Unique, volet données personnelles
- mars 2008 : mise en production du Dossier Fiscal Unique, volet données fiscales, contrôle, perception et recouvrement.

Ce projet est réalisé avec l'aide d'un consortium de sociétés externes qui assistent le S.P.F. Finances dans l'implémentation des solutions élaborées.

Les ressources affectées à ce projet sont les suivantes :

- un directeur de projet internes et un directeur de projet externe ;
- quatre chefs de projet interne et quatre chefs de projets externes affectés respectivement aux quatre modules que sont le volet données personnelles du Dossier Unique, le volet données fiscales du Dossier Unique, le volet contrôle et le volet perception et recouvrement ;
- dix analystes « métiers » internes et huit analystes externes ;



- quatre groupes de travail (un par module) constitués chacun d'une vingtaine d'experts « métiers » du S.P.F. Finances.
- deux architectes internes et trois architectes externes ;
- une dizaine de testeurs internes et externes ;
- cinq développeurs internes et une vingtaine de développeurs externes ;
- une cellule de gestion du « change management » (quatre personnes) ;
- un staff d'administration et de coordination de projet (six personnes) ;

De façon ponctuelle et selon les besoins du projet, le S.P.F. Finances met à la disposition du projet toutes les ressources spécialisées nécessaires. Il s'agit surtout ici de ressources IT tels que les administrateurs de bases de données, les responsables de l'infrastructure informatique, les responsables de l'infrastructure applicative ....

En tout, plus de cent soixante personnes affectées à la réalisation d'une application informatique qui sera utilisée par quelque trente mille fonctionnaires.

### 4.2.3. Risques

#### La gestion centralisée des fiches cadastrales numérisées

- acceptation des nouvelles méthodes de travail par les personnes affectées au projet ;
- gestion d'un projet uniquement en interne sans aide extérieure ;
- formation insuffisante des ressources aux nouveaux standards (outils de modélisation, maîtrise du langage UML, technique de tests, ...) ;
- inadéquation de la méthodologie avec le type de projet (inadaptation des efforts demandés en regard de l'importance du projet) ;
- sous-estimation des ressources nécessaires au projet ;
- la mise en place de la nouvelle organisation et des nouvelles fonctions n'est pas effective.

#### Système de Traitement Intégré, Impôts et Recouvrement (STIR).

- impact de la mise en place de la méthodologie sur les plannings établis préalablement ;
- impact de la mise en place de la méthodologie sur les budgets établis préalablement ;
- non adhésion du consortium de société externes impliquées dans le projet à la mise en place de nouveaux standards méthodologiques non explicitement prévus dans le cahier des charges initial ;
- formation insuffisante des ressources internes au S.P.F. Finances aux nouveaux standards (outils de modélisation, maîtrise du langage UML, technique de tests, ...) ;
- découverte de manquement entre la méthodologie et son application sur le terrain ;
- projet trop important pour être projet « pilote » ;
- la mise en place de la nouvelle organisation et des nouvelles fonctions n'est pas effective.

#### 4.2.4. Contraintes

##### La gestion centralisée des fiches cadastrales numérisées

- ressources limitées en personnel ;
- pas de possibilité de faire appel à des ressources externes ;
- projet non prioritaire dans le plan stratégique du S.P.F. Finances ;

##### Système de Traitement Intégré, Impôts et Recouvrement.

- budget fixe ;
- plannings établis avant la décision d'implémentation méthodologique entraînant un besoin de synchronisation important entre le projet SupDev et le projet STIR ;
- engagements contractuels avec le consortium de sociétés ayant remporté le marché public STIR non nécessairement en ligne avec les préceptes imposés avec la nouvelle méthodologie;

#### 4.2.5. Etat d'avancement des projets pilotes

##### La gestion centralisée des fiches cadastrales numérisées

###### **Analyse métier:**

Le Modèle métier

###### **Le Modèle des processus métier**

- diagramme des classes acteurs (100%)
- diagrammes de cas d'utilisation métier (100%)
- descriptions textuelles des cas d'utilisation métier (80%)
- diagrammes d'activité (60%)
- diagrammes de séquence (facultatifs)

Le Modèle du domaine ou « Modèle des entités métier»

- diagramme de classes modélisant les entités métier (80%)
- diagrammes d'états – transitions pour certaines entités métiers (60%)

Le Glossaire, qui est transverse à tous les projets et à tous les processus métier. (100%).

###### **Exigences :**

Le document Vision reprenant une synthèse des besoins, le contexte du projet, le périmètre du projet et dont l'objectif est de permettre à tous les acteurs du projet d'obtenir une compréhension claire et rapide de celui-ci (100%)



Le document spécifications supplémentaires décrivant les exigences non fonctionnelles, souvent transversales et applicables à tous les cas d'utilisation (Interface utilisateur, architecture technique, performance,...) (10%)

La liste complète des exigences recensées à partir des cas d'utilisation pour les exigences fonctionnelles et du document spécifications supplémentaires pour les exigences non fonctionnelles (10%)

### **Analyse et Conception:**

Le dossier d'**architecture logicielle** (60%)

Le modèle interface utilisateur comprenant :

- Les maquettes statiques des écrans (10%)
- Les modèles cinématiques de l'interface utilisateur (diagramme d'activité) (40%)

Les modèles des données comprenant :

- Le modèle conceptuel des données (diagramme de classe) (80%)
- Le modèle physique des données (diagramme entité/relation) (70%)

Le modèle d'analyse et conception comprenant

- Le diagramme des classes participantes
- Les diagrammes d'interaction (séquence et/activité)

Les **diagrammes de robustesse** (facultatifs)

### *Système de Traitement Intégré, Impôts et Recouvrement.*

Ce projet est assez complexe dans sa gestion et son administration.

En effet, il est composé de quatre modules indépendants dans leur développement mais très liés quant à leurs interdépendances fonctionnelles et techniques. Les développements liés à ces modules ne sont pas nécessairement parallélisés et l'état d'avancement diverge fortement de l'un à l'autre. De plus certains éléments de développement sont commun (Use Cases communs, architecture technique).

Afin de ne pas alourdir inutilement ce mémoire, nous ne tirerons que les lignes directrices de l'état d'avancement du projet.

De manière générale, nous pouvons constater que tous les éléments transversaux de la méthodologie sont parcourus (glossaire, dictionnaire des données, conventions de nommage).

Quant aux différentes disciplines de la méthodologie FUP, elles ont été au moins une fois entièrement parcourues.

Actuellement le module d'implémentation du Dossier Unique, volet données personnelles, est en phase de test d'acceptation pour les migrations et les gestions d'anomalies.

## **5. Les premiers enseignements retirés des projets pilotes en cours**

Dans cette section nous relèverons les principaux enseignements retirés du déroulement des projets pilotes. Ces différents aspects ont été relevés par le Core Team SupDev auprès des différents acteurs impliqués dans les projets pilotes. Nous avons voulu dans cette section faire la synthèse des réactions qui nous sont parvenues, une analyse de leurs impacts et des mesures correctives apportées.

Il ne s'agit donc pas d'une évaluation objective de l'ensemble du projet SupDev, évaluation qui fera l'objet de la section suivante.

### **5.1. Aspect Méthode**

#### **5.1.1. Le modèle itératif**

Force a été de constater que les concepts de méthode itérative et incrémentale ne sont pas assez assimilés par les équipes de développement, et ceci aussi bien au niveau des analystes qu'au niveau des développeurs.

Cet état de fait s'est surtout rencontré dans le cadre du projet interne « gestion centralisée des fiches cadastrales numérisées » pour lequel seulement des agents du S.P.F. Finances sont impliqués.

Nous avons en effet remarqué une forte tendance à mettre en place un système en cascade prenant à la lettre le cycle de vie défini dans FUP mais en passant sous silence l'aspect itératif.

Il s'ensuit donc que les démarches de construction, d'acceptation, de tests et d'implémentation sont souvent ignorés tant que les analyses détaillées ne sont pas terminées pour l'ensemble de la future application.

L'analyse plus poussée de ce phénomène soulève également le manque de guidance dans la gestion du projet par manque de repère méthodologiques appropriés et par manque de formation des chefs de projet.

Ces repères ne font pas partie de la démarche méthodologique de FUP mais leur absence influence grandement celle-ci. Nous reviendrons sur ce point lorsque nous parlerons des aspects humains et des aspects « projet ».

Au niveau du projet d'investissement STIR, l'aspect itératif et incrémental du développement est beaucoup mieux assimilé et respecté.

Ceci est certainement la conséquence du recours à des ressources externes au S.P.F. Finances bien mieux formées et habituées à ces techniques de développement, mais aussi plus aguerries au niveau de la gestion de projet adapté à la méthodologie de développement.



### 5.1.2. Le cycle de vie

L'élaboration du cycle de vie général adopté par la méthodologie FUP prévoyait la découpe d'un projet en deux phases bien définies : la préétude et le projet en lui-même.

La vision dans laquelle cette décision a été prise était celle des développements futurs d'application, axée sur les nouvelles technologies. Sous-entendu à ceci que ces projets seraient soumis à appels d'offre, ce qui entraînait une phase de préétude pour la rédaction du cahier des charges.

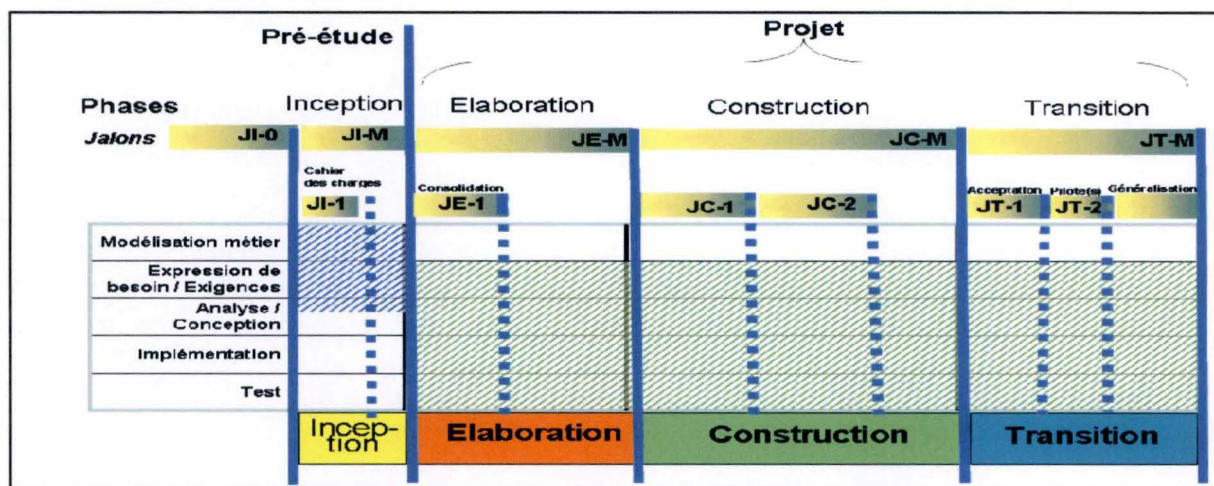


Figure 5-1 : Le cycle de vie d'un projet externalisé

Dans le cadre du projet interne « gestion centralisée des fiches cadastrales numérisées », il est apparu que cette découpe préétude/projet ne répondait pas aux besoins. Cependant l'élaboration d'un cahier des charges n'était pas dépourvue d'utilité dans le cadre de tel projet.

La phase d'Inception doit donc alors s'interpréter comme une étude de faisabilité concrète produisant un livrable reprenant la modélisation métier de haut niveau, l'expression des besoins et des exigences ainsi qu'une ébauche d'architecture fonctionnelle et technique. Ce livrable sera présenté au Comité de Pilotage et au Comité de Direction qui statueront alors sur un GO/NOGO pour la phase de développement du projet.

La méthodologie FUP devra donc être adaptée pour prendre en charge cet aspect « Cycle de vie » et pour formaliser les livrables nécessaires à l'accomplissement de la phase Inception.

Dans le cadre du projet STIR, nous nous trouvons déjà dans la phase projet puisqu'une préétude avait déjà été réalisée et avait produit un cahier des charges qui avait conduit à un appel d'offres.

### **5.1.3. Les livrables**

La méthodologie a pour but de guider le développement en définissant la méthode de travail et en guidant les différents intervenants dans les tâches à accomplir.

Parallèlement à celle-ci, la suite logicielle retenue pour le développement aide et guide également l'utilisateur dans son travail.

La philosophie de ses outils est que toutes les informations doivent être entrées dans les logiciels et que c'est à travers eux que la documentation imprimée (les livrables) est fournie.

Contrairement à une suite logicielle comme les outils Rational qui acceptent d'associer des fichiers externes (traitement de texte, feuilles de calculs), ces pratiques sont interdites dans le cadre de SupDev.

Il s'ensuit donc que les livrables à fournir lors du développement doivent avoir préalablement été définis quant à leur structure dans les outils.

Il a été constaté à travers les projets pilotes que bon nombre de ces livrables n'avaient pas été prévus soit parce qu'ils n'avaient pas été formatés dans l'outil, soit que le « Core Team » n'avait pas pensé à la pertinence de leur emploi.

Une seconde constatation est également apparue : pour que les livrables soient exploitables pour chaque projet, ceux-ci doivent respecter un certain formalisme dans l'utilisation des outils. Pour ce faire, chaque projet devrait respecter une certaine architecture dans l'élaboration de ses modélisations.

Afin d'officialiser ce formalisme, un mode opératoire précis sera mis en place.

### **5.1.4. Le Dictionnaire des données**

Dans la méthodologie FUP, il avait été prévu de mettre en place un dictionnaire transversal des données manipulées par les projets. L'idée principale de la transversalité était qu'une donnée ne devait apparaître qu'une seule fois aussi bien dans les classes que dans les bases de données.

Ce principe devait assurer la réutilisabilité des concepts métier (classe) d'un projet à l'autre.

Il est apparu que l'utilité de cette démarche quoique avérée ne donnait pas entièrement satisfaction. En effet, dans le monde orienté objet, plus que la donnée, c'est l'objet qui est au centre de toutes les attentions ; l'objet avec ses méthodes, son constructeur et les données qui lui sont associées.

Face à ce constat, le « Core Team » a mis sur pied un groupe de travail spécifique afin de mettre au point un référentiel transversal des objets, présentant ceux-ci et leurs méthodes associées. L'idée étant encore une fois d'assurer et de promouvoir la réutilisabilité des concepts « métier ».

Ce travail est actuellement en cours de réalisation et met en évidence quelques principes primordiaux tels que la bonne documentation mise continuellement à jour et le principe de non-régression des développements.



## **5.2. Aspect changement**

### **5.2.1. Besoin de formation**

La mise en place d'une nouvelle méthodologie nécessite un effort d'apprentissage non négligeable de la part de tous les participants à un projet. Nous avons prévu une formation à la méthodologie axée plus spécifiquement sur les disciplines assignées aux intervenants selon le principe de « chacun sa spécialité ».

Cependant, le cloisonnement des formations en fonction des rôles et fonctions a mis à jour pas mal de dysfonctionnements dans la compréhension globale de la méthodologie.

Le principal inconvénient de cette façon de procéder est que personne ne maîtrise les disciplines dans leur ensemble, voire ne comprend pas le travail de l'autre. Ce manque de compréhension générale est sans nul doute à la base du manque de maîtrise du concept de développement itératif et incrémental.

De plus cela crée un cloisonnement dans les différents métiers liés au développement qui s'avère vite devenir un frein à la communication et au travail d'équipe.

Un effort particulier devra être entrepris dans le cadre de la formation pour assurer une culture générale de la méthodologie avant d'aborder les spécificités propres à chaque discipline.

De plus, il a été décidé d'assurer une formation continue des agents par la mise en place de cursus de mise à niveau afin d'assurer l'ancrage de la méthodologie et de ses évolutions.

### **5.2.2. Transfert de connaissances et d'expériences**

Les équipes actuellement en place dans les deux projets pilotes sont assez isolées les unes des autres et se parlent relativement peu.

Ce manque de communication a pour conséquence que ces équipes échangent très peu leurs expériences et leurs connaissances, retardant d'autant plus l'acquisition optimale des principes édictés par SupDev.

Afin de pallier cet état de fait, des réunions de coordination et de consolidation mensuelles ont été mises sur pied afin de remédier à ce manque d'échanges. Ces réunions sont menées par une équipe de formateurs à la méthodologie. Le résultat de ses réunions est mis en ligne sur l'intranet sous forme d'un FAQ mais aussi sous forme d'un forum de discussions thématiques. Le succès rencontré par ce dispositif montre à souhait le besoin qu'ont les participants aux projets d'échanger leurs expériences, de transférer leurs connaissances et d'assurer leur formation continue.

## **5.3. Aspect humain**

### **5.3.1. Absence de structures organisationnelles soutenant les processus de développement de FUP**

L'absence d'opérationnalisation effective des fonctions et de l'organisation dans la vision de CoperFin plonge les agents du S.P.F. Finances dans un environnement instable et insécurisant.

En effet, la réforme des méthodes de travail relatives au développement de projets axés sur les nouvelles technologies prévoyait la mise en place d'une nouvelle organisation soutenue par des nouvelles fonctions. La méthodologie FUP s'axe sur cette nouvelle organisation.

De cet état de fait découle que les agents affectés aux projets pilotes se trouvent confrontés à deux mondes différents et ne savent pas exactement comment se positionner. D'une part, l'organisation actuelle ne prévoyait pas les processus de travail initiés par la méthodologie FUP et n'est pas à même de les soutenir. D'autre part, l'organisation TO-BE est encore virtuelle pour le S.P.F. Finances mais les fonctionnaires affectées aux projets pilotes doivent s'y conformer.

La coexistence de ces deux organisations n'est pas efficiente.

Ce problème a été remonté au Comité de Direction qui en accord avec le service P&O (personnel et organisation) va devoir prendre la décision d'opérationnaliser les principes fonctionnels et organisationnels de CoperFin.

### **5.3.2. Suivi plus proche des ressources**

A la suite d'entretiens menés avec les agents affectés aux développements des projets pilotes, il ressort que ceux-ci ont le sentiment d'avoir un peu « été lâchés dans la nature » sans qu'un support et un suivi n'aient été prévus.

Pour répondre à cette demande, une cellule de « soutien méthodologique » a été mise en place (sorte de helpdesk aussi bien d'un point de vu méthodologique qu'au niveau de l'utilisation des outils logiciels de développement).

De plus, les réunions de coordination et de consolidation dont il est fait état dans le chapitre 5.2.2 ont répondu en grande partie aux attentes de ces agents.

### **5.3.3. Mieux exprimer les enjeux stratégiques de l'adoption d'une méthodologie**

Lors de l'évaluation des formations octroyées aux agents impliqués dans les projets pilotes, nous avons rapidement constaté que les enjeux stratégiques de l'adoption d'une méthodologie n'étaient pas compris. Ceci entraînait un manque de vision des objectifs fixés, mais aussi un manque d'adhésion du personnel à la démarche.

Pour y remédier, nous avons eu recours à deux types de solutions :



- 1- mise en place lors de formations d'une demi journée afin d'exposer les enjeux stratégiques et les changements induits par CoperFin en matière de développement informatique.
- 2- établissement d'un plan de communication interne auprès des analystes et des intervenants ICT d'un projet informatique. Ce plan est actuellement mis en application et des journées de présentation sont organisées dans les différents départements.

Le retour de ces initiatives est très positif, non seulement auprès des personnes directement intéressées mais aussi auprès des intervenants voisins ou gravitant autour de projets.

## **5.4. Aspect projet**

### **5.4.1. Besoin d'une méthodologie de gestion de projet**

Le déroulement du développement d'un projet doit être piloté par un chef de projet formé aux techniques de gestion de projet.

Au sein du S.P.F. Finances cette culture de gestion de projet n'est pas acquise et entraîne beaucoup de difficulté pour mener à bien les missions. Non seulement la gestion est bicéphale (il y a un chef de projet métier et un chef de projet ICT) mais ceux-ci sont généralement choisis parmi les analystes et les spécialistes métier ou ICT peu au fait des techniques de gestion de projet.

Les conséquences sont multiples : confusion des rôles et des tâches, établissement de plannings non réalistes et non tenues de ceux-ci, dépassement budgétaire, objectifs de périmètre de projet non atteints, etc....

Les managers du S.P.F. Finances ont pris conscience de cet état de fait et ont initié des actions pour y remédier :

- ↳ Mise en place d'une méthodologie de gestion de projet (Prince 2) au travers d'un projet d'implémentation similaire à SupDev.
- ↳ Recrutement de personnel qualifié au profil de chef de projet au travers du Sélor (organisme de recrutement des agents de l'Etat)
- ↳ Mise en place de PMO (Program management office) afin de mieux suivre et aider les projets en cours.

Les effets de ces mesures ne sont pas encore mesurables vu leur mise en application récente, mais il apparaît déjà que le recours au professionnalisme du rôle de chef de projet évite pas mal de dérives.

### **5.4.2. Besoins d'un référentiel modélisé des standards de développement IT**

Lorsque le projet pilote « Gestion centralisée des fiches cadastrales numérisées » à débiter la détermination de son architecture technique, il s'est retrouvé confronté à devoir lui-même modéliser les principes ICT standards du S.P.F. Finances tel le Framework CCFF (voir 3.2.1) ou encore les interactions avec les autres projets en cours de développement.

Le problème ne résidait pas uniquement dans la charge de travail demandée pour cette modélisation, mais surtout dans le fait de la non maîtrise des technologies et concepts manipulés.

En effet, le standard CCFF par exemple est très complexe dans son fonctionnement mais son utilisation et le recours à ses services doivent se faire au travers d'interfaces de communication. Ces interfaces devraient déjà être définies au sein de l'organisation et être offertes comme des modules utilisables par les autres projets.

Pour le bon fonctionnement de l'ensemble des projets utilisant CCFF (projet nouvelle technologie) un référentiel modélisé de ses services doit exister de manière transversale à toute l'organisation.

Suite à cette constatation, il a été décidé que tout projet de standardisation devait effectuer ce travail afin d'assurer la cohérence de ses services et réduire l'impact des changements et adaptations futures de ceux-ci (non-régression, réutilisabilité, extension).



## **6. Recommandations à l'implémentation de la méthode au S.P.F Finances**

Dans cette section, nous traiterons des aspects qui nous semblent essentiels d'aborder et de résoudre au sein du S.P.F. Finances afin de permettre une implémentation optimale de la méthodologie FUP. Ces différents aspects ont fait l'objet de discussions au sein du Comité de Pilotage du projet SupDev. Des décisions correctives ou adaptatives ont été prises pour certains aspects ; pour d'autres, le Comité de Direction a été mis au courant afin d'exercer son pouvoir de décision stratégique.

### **6.1. Aspect organisation : mettre en application effective la réforme CoperFin.**

La connaissance de l'entière de l'organisation dans laquelle une méthodologie de développement doit être implémentée est primordiale.

Les résultats de l'Assessment réalisé auprès des services ICT du S.P.F. Finances a apporté beaucoup d'informations mais n'a pas pu cerner l'entière des intervenants à la réalisation d'un projet informatique. En effet, historiquement, les développements de programmes étaient confinés dans les services ICT et l'intervention des services « métiers » lors de l'analyse des besoins était réduite à leur plus simple expression. Lors de l'Assessment mené durant la phase d'analyse du projet, SupDev a sous-estimé cette caractéristique et n'a pas assez pris en compte l'organisation et l'implication des intervenants « métier ».

Cependant, on pourrait contredire cette observation en mettant en avant la réforme CoperFin en cours au S.P.F. tant dans son aspect organisationnel que fonctionnel. L'administration est en pleine restructuration et ses processus de travail changent. La vision à venir (TO-BE) de l'organisation du travail passe par la refonte des fonctions, des tâches et des services.

Le projet d'implémentation méthodologique SupDev a dû se conformer à cette vision TO-BE de l'organisation, mais sa mise en œuvre se déroule dans un environnement non encore réformé, autrement dit non adapté à l'organisation appropriée pour assurer le bon déroulement de sa mission.

Comme nous l'avons déjà signalé plus avant, les conséquences humaines de cet état de fait sont nombreuses : manque d'enthousiasme, non-adhésion, incompréhension des objectifs stratégiques.

Ceci montre à souhait que le binôme « mise en production/organisation » est primordial afin d'assurer l'efficacité du travail à accomplir et de l'assertivité des personnes impliquées.

#### **Recommandations :**

Le S.P.F. Finances devrait veiller à l'opérationnalisation de la réforme CoperFin en fonction des réalisations de projets en cours en son sein.

Cette opérationnalisation devrait se dérouler de manière itérative et incrémentale, partant des projets pilotes et se poursuivant durant la mise en production en masse des outils soutenant les nouvelles méthodes de travail.



## **6.2. Aspect changement**

### **6.2.1. Eviter le Big Bang.**

La mise en place de la méthodologie FUP au travers des projets pilotes démontre déjà largement son utilité et sa nécessité dans les développements. A tel point que la tentation est forte du côté des chefs de projet de basculer le mode de fonctionnement de leur propre projet vers cette méthodologie.

L'idée en soi n'est pas critiquable quant il s'agit de nouveaux projets orientés vers les nouvelles technologie. En effet, ces projets doivent utiliser les nouveaux standards de l'ICT du S.P.F. Finances et répondent donc pleinement avec la philosophie de la méthodologie FUP. En plus, la tenue des différents glossaires et dictionnaires transversaux les obligent déjà à adopter quelques principes standardisés de la méthodologie.

Cependant, basculer ainsi un nombre important de projets vers FUP présente actuellement de grands risques. Outre le fait que l'organisation n'est pas encore structurée pour accueillir en masse ces changements de méthodes de travail, les ressources humaines ne sont pas non plus opérationnelles car non formées aux rôles et fonctions décrites dans FUP, non accoutumées au développement sur les outils de modélisations adoptés par le S.P.F. ou encore non habituées aux principes fondamentaux de la méthodologie.

De plus, FUP en est encore à sa phase pilote et nécessite toujours des adaptations et des affinements dans ses processus de travail. Mais après une phase de consolidation et une validation de son entièreté, l'extension de la méthodologie pourra se faire sur les nouveaux projets dès l'initiation de ces derniers.

Mais qu'en sera-t-il alors des projets déjà en développement. Le Comité de Direction du S.P.F. a décidé que les grands projets d'investissement à forte implication métier et étalés sur plusieurs années doivent basculer progressivement vers la méthodologie FUP. Il s'agit de projets tels le STIR dans sa totalité, STIPAD (système de traitement intégré de l'administration patrimoniale et domaniale) ou encore PLDA (Paperless Douane, système de traitement intégré de la Douane) qui devront, une fois terminés, ne former qu'une seule et même application. Ceux-ci comptent pour environ 1/5 du nombre total de projets en cours au S.P.F., mais représentent 85% des budgets engagés.

Le choix de ses projets n'est pas neutre. En effet, ils ont tous été soumis à la procédure d'appel d'offre et des sociétés externes sont impliquées dans le développement. L'aide de ces externes et leur connaissance des principes méthodologiques liés à FUP rendent cette démarche réaliste et réalisable.

Le basculement progressif des « gros » projets déjà existants à la méthodologie FUP et l'intégration progressive des nouveaux projets à venir FUP permettront l'implémentation en douceur de celle-ci en évitant un big bang ingérable.



### 6.2.2. Veiller à la cohérence du contrôle, de la gestion et du développement d'un projet

L'implémentation d'une méthodologie de développement ne doit pas être un acte isolé sans accroches fermes avec un ensemble cohérent et structuré de gestion organisationnelle.

En effet, ignorer les concepts fondamentaux comme la gestion de projet ou encore le contrôle et le suivi managérial des portefeuilles de projets et de programmes ne pourra que mener à la faillite de l'organisation.

Les changements induits par la structuration de toute la chaîne de conduite des projets sont nombreux et profonds. Les BPR CoperFin ont mené à l'idée de création de PMO's (Program management office) qui seront attachés à nos managers tant au niveau N (niveau stratégique) qu'au niveau N-1 (niveau opérationnel). Les processus de travail ont été établis et un groupe de réflexion est actuellement en charge d'opérationnaliser ces nouveaux organes d'aide et de contrôle des programmes et projets.

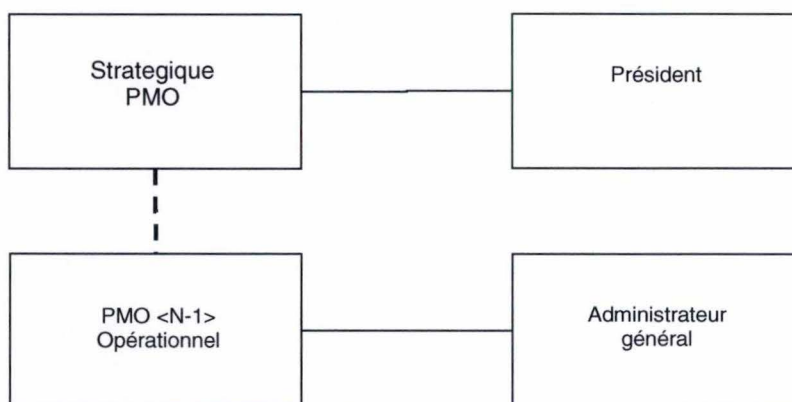


Figure 6-1 : Organisation des PMO's

Pour rappel, un programme est un ensemble de projets qui convergent vers un même objectif « métier » et qui dépendent d'un même pilier fiscal.

Au niveau de la gestion de projet, l'implémentation de la méthodologie Prince 2 est actuellement en cours de réalisation. Un projet spécifique, appelé « Gestion de projet » et dont le mode de fonctionnement est comparable à celui de SupDev a vu le jour en ce début d'année. Il s'agit d'implémenter une méthodologie soutenue par une suite logicielle appropriée. (Mise en production de la méthodologie Prince 2 planifiée pour début 2007).

Visuellement, les interdépendances entre les trois concepts (contrôle, gestion et développement) peuvent se voir comme dans la figure suivante.



Figure 6-2 : Contrôle, gestion et développement d'un projet

Nous devons cependant constater que la planification de réalisation de ces différents projets n'est pas synchronisée, ce qui conduit à un manque de vision générale du changement et à une inconsistance générale de la méthode globale.

L'idéal aurait été d'adopter un traitement top-down de l'implémentation méthodologique, commençant par le contrôle général, ensuite implémenter la gestion et terminer par le développement. La réalité nous montre que la planification est inverse et adopte une approche Botton-up.

Conscient des incohérences de la démarche, le Comité de Direction du S.P.F. Finances a décidé de hâter la réalisation effective des PMO et de la gestion de projet en parallélisant leur développement et en synchronisant leur mise en production.

Cette décision est lourde de conséquences sur l'aspect management du changement au sein de notre organisation car elle cumule tous les risques inhérents à l'implémentation de nouvelles méthodologies.



## **6.3. Aspect humain**

### **6.3.1. Renforcer la gestion des ressources humaines**

La découpe de la méthodologie en discipline métier avec l'assignation de rôles et fonctions a provoqué une forte demande de ressources différents et spécialisées ; contrairement aux développements antérieurs qui demandaient moins de ressources différentes mais employaient celles-ci à temps plein et exigeaient d'elles une très forte polyvalence.

La conséquence de cet état de fait est qu'il faut absolument renforcer les équipes par projet tout en assurant une gestion optimale des ressources sur plusieurs projets. En effet, le travail assigné à un testeur ou à un développeur par un projet ne l'occupe pas à cent pour cent durant toute la durée de celui-ci.

Cette gestion de l'affectation des ressources n'est pas prévue dans la méthodologie FUP et doit revenir à la méthodologie de gestion de projet avec l'aide appuyée des PMO. Cependant, vu l'absence opérationnelle de ces deux projets, il faut instaurer ce mode de gestion de manière transitoire.

### **6.3.2. Recruter des ressources avec des profils professionnels définis**

La demande accrue prévisible de ressources à affecter aux projets qui « basculeront » dans la nouvelle méthodologie oblige le S.P.F. Finances à procéder au recrutement de personnel tant au niveau interne qu'externe.

La professionnalisation et la spécification des rôles et fonctions à remplir entraînent la nécessité de définir les profils recherchés afin d'arriver à une entrée en fonction efficace le plus rapidement possible.

Antérieurement, la politique menée en matière de recrutement était de sélectionner sur base du niveau d'études (universitaire, graduat, secondaire supérieur) et d'assurer la formation des nouvelles recrues en fonction des besoins internes spécifiques.

La professionnalisation des profils d'experts ICT et d'analystes « métier » ne permet plus de partir de zéro et de former sur le tas. Ainsi pour renforcer les cadres, le S.P.F. Finances recrute via le SELOR (organisme de recrutement des agents de l'Etat) les profils recherchés. Ainsi on prévoit pour septembre 2006 l'entrée en fonction d'une dizaine de chefs de projet, d'une vingtaine d'analystes et d'autant de testeurs, développeurs et administrateurs système.

Cependant, il ne faudrait pas perdre de vue que la durée d'un projet est limitée dans le temps et qu'une fois que les projets qui sont liés à la réalisation des objectifs de CoperFin seront réalisés, le nombre de ressources nécessaires au développement et à la gestion de projets ne sera plus aussi élevé.

Ainsi l'idée de faire appel à des ressources temporaires est-elle à l'étude actuellement. Il serait fait appel à des sociétés spécialisées dans la mise à disposition de ressources spécifiques (souvent à profil ICT) vers qui on se tournerait pour renforcer temporairement le noyau dur du S.P.F.

Cette idée est à l'étude et devra recevoir l'aval du comité de Direction, du Conseil des Ministres et passer auprès des organisations syndicales. Un long chemin en perspective dont l'issue est encore inconnue.

### **6.3.3. Veiller à la formation continue des différents acteurs**

La méthodologie FUP a été voulue dynamique et évolutive. Ainsi au fur et à mesure de son utilisation elle évoluera et s'adaptera aux exigences et spécificité de notre organisation.

Afin que les intervenants aux projets puissent suivre cette évolution et acquérir de manière optimale les changements induits, un système de formation continue doit être mis en place. Il ne s'agit pas uniquement ici d'assurer la formation des seuls agents internes au S.P.F. Finances, mais aussi celle de tous les intervenants externes impliqués dans les projets au travers des soumissions aux cahiers des charges.

A noter que les intervenants externes sont sensés maîtriser les standards du S.P.F. (dont la méthodologie FUP fait désormais partie) lors du commencement de la réalisation du projet qui leur a été confié. Cependant, le S.P.F. doit veiller à assurer le suivi et l'acquisition des modifications et adaptations de ces standards.

## **6.4. Aspect projet**

Les deux sujets abordés ci-dessous ne sont pas directement pris en charge par la méthodologie FUP, mais touchent directement le développement de projet. Ce sont ce que l'on pourrait appeler des effets de bord initiés par la philosophie de FUP et qui doivent être pris en charge par la gestion de projet et la mise en place des PMO's.

### **6.4.1. Assurer le partage des informations transversales entre les projets**

Dans le cadre de FUP, pour des soucis d'harmonisation, de réutilisation et de standardisation, il a été mis en place des concepts transversaux tels le glossaire, les conventions de nommage ou encore le dictionnaire des données.

Un point important a en plus été relevé lors des premiers enseignements issus de la réalisation des projets pilotes : la modélisation des standards techniques de l'ICT et l'utilisation de ceux-ci au travers de tous les projets « nouvelles technologies ».

L'évolution permanente des standards oblige à une mise à niveau constante des connaissances de chaque projet impliqué par ces modifications. L'évolution technologique doit aussi être répercutée sur ces différents projets.

Afin de mieux assurer le partage de ses informations, le S.P.F. Finances a mis sur place différents comités inter-projets. Le déroulement de ces comités auxquels un intervenant de chaque projet est convié permet de faire passer l'information de manière plus formelle. C'est aussi l'occasion pour chacun d'exposer ses demandes aux autres projets et de coordonner les solutions techniques et « métier » adoptées.

Ainsi, la vision de chacun est-elle identique et unanimement reconnue.



### **6.4.2. Synchroniser les plannings**

La gestion des dépendances et des responsabilités inter-projets n'est normalement pas assurée par la méthodologie FUP mais rentre normalement dans le périmètre de la gestion de projet et celui des PMO's.

Cependant, FUP instaure le principe qu'un concept n'est modélisé et développé qu'une seule fois au sein de l'organisation. Les projets qui voudraient utiliser à leur tour ces concepts doivent user alors des modélisations déjà développées. Ceci sous-entend aussi la connaissance inter-projets des développements en cours de réalisation. Cette connaissance est assurée par la tenue des comités évoqués dans la section précédente et par la mise en commun des modélisations par la suite des logiciels de modélisation.

Mais une dimension manque encore : les plannings de réalisation.

La gestion des plannings est encore actuellement confinée à chaque projet et leur mise en commun n'est pas assurée de manière centralisée. Le projet « Gestion de projet », de par la méthodologie et la suite logicielle qu'il mettra en place, répondra à ce besoin.

De manière transitoire, cette gestion est centralisée auprès de deux agents des Finances qui ont pour mission de contacter tous les chefs de projets pour établir ces dépendances entre projets et synchroniser les plannings.

## **7. Evaluation du projet SupDev en fonction des critères de réussite définis précédemment (section 3.4)**

Dans cette section nous procèderons à l'évaluation objective du projet SupDev dans sa phase d'implémentation auprès des projets pilotes. A ce stade de réalisation du projet, tous les objectifs ne sont pas atteints ou en tout cas ne sont pas actuellement quantifiables au niveau de leur réalisation.

### **7.1. Mettre en place un standard imposé à tous les intervenants**

Grâce à la volonté du management et des différents niveaux de hiérarchie intermédiaire, l'adoption du standard méthodologique FUP et des outils de modélisation est effective. Cette affirmation doit cependant être nuancée en fonction de l'évolution de développement des projets concernés comme nous l'avons exposé dans la section 6.2.1 (Eviter le Big Bang).

A terme, cet objectif sera assurément atteint et la mission de SupDev correctement remplie.

### **7.2. Maîtriser les coûts de développement**

Ce critère de réussite est actuellement difficilement évaluable dans ce sens que les projets pilotes ne nous permettent pas de tirer des enseignements et des conclusions fiables.

Pour le projet « gestion centralisée des fiches cadastrales numérisées », aucun budget n'a été débloqué pour sa réalisation dans le sens où il s'agit d'un projet interne. Actuellement, nous ne disposons pas d'outil d'audit interne, comme une comptabilité analytique, pour évaluer le coût réel d'un tel projet.

Concernant le second projet « pilote », STIR, les résultats d'analyse budgétaire montrent une augmentation des coûts liés à l'adoption de la méthodologie en cours de développement. Il a en effet fallu que le projet s'adapte aux concepts généraux de FUP, ce qui a engendré un travail supplémentaire.

Par contre, il semble acquis que le recours aux standards transversaux (glossaire, dictionnaires des données, application des conventions de nommage) engendre une forte diminution de la charge de travail lors du développement d'application utilisant des concepts déjà définis.

De manière générale, nous pouvons affirmer que si l'adoption des standards méthodologiques n'a pas encore entraîné de substantielles économies, les projections dans le futur sont optimistes.

### **7.3. Mettre en place un cadre strict et porteur**

Le projet SupDev a publié ses référentiels méthodologiques et ses modes opératoires. Ces documents sont en cours de validation formelle et une fois celle-ci effectuée, ils seront mis à la disposition de tous les intervenants aux projets en cours au S.P.F. Finances et seront déclarés référents obligatoires.

Cet objectif assigné au projet SupDev est en cours d'être atteint.



## **7.4. Assurer la polyvalence**

Comme nous l'avons expliqué plus avant, FUP n'est pas une méthodologie définie une fois pour toute. Au contraire elle se veut vivante et en continuelle évolution en fonction des besoins initiés par les projets et en rapport avec l'évolution qualitative de notre organisation.

FUP est actuellement une méthodologie généraliste ne distinguant les différents projets qu'en fonction de leur taille. Nous sommes conscients que c'est trop restrictif.

Afin d'atteindre l'objectif de polyvalence fixé, le Core Team travaille de manière continue à l'évolution de la méthodologie.

Pour une analyse plus approfondie sur ce sujet, nous renvoyons le lecteur à la section 8 « Evolution de la méthodologie ».

## **7.5. Maîtriser toutes les étapes et les productions**

Ce critère de succès n'est pas encore atteint malgré les efforts de formations entrepris par le S.P.F. Finances. Nous avons déjà traité ce sujet dans la section 5.1 « Aspect méthode » dans la partie traitant des premiers enseignements retirés auprès des intervenants aux projets.

Un effort particulier est mis en place au niveau des formations de base et des formations continues.

Cet objectif sera réévalué de manière constante durant l'implémentation de la méthodologie à toute l'organisation.

## **7.6. Assurer la prise en compte des aspects « métier » du développement**

Cet objectif est pleinement atteint. En effet la méthodologie FUP est axée sur une approche « discipline métier » du développement de projet. Les livrables sont construits également autour de cette approche.

## **7.7. Assurer la maîtrise d'une suite logicielle permettant le soutien de la méthodologie par l'élaboration de modes opératoires précis**

Les modes opératoires principaux sont construits et les formations aux outils sont continues. La maîtrise de la suite logicielle par les différents intervenants aux projets s'accroît au fur et à mesure de l'utilisation de celle-ci.

Un audit interne de qualité montre que le niveau de compétence dans les outils est en constante progression.

## **7.8. Assurer une évolution qualitative des développements**

Selon un rapport remis par une société externe dans le cadre du « Quality Control » des projets en cours de développement au sein du S.P.F. Finances, l'évolution qualitative des développements des projets utilisant les outils de modélisation préconisés par SupDev et appliquant les modes opératoires définis par lui, est positive.

La documentation est plus complète, les analyses plus structurées et mieux finalisées, les architectures mieux définies et standardisées, les modélisations UML plus complètes et mieux exploitées.

Cette évolution doit être constante et sous contrôle, mais nous pouvons affirmer que l'objectif qualitatif de SupDev est atteint.

## **7.9. Assurer l'évolution qualitative de l'organisation**

Cet aspect est difficilement évaluable dans le sens où l'implémentation d'une méthodologie de développement ne représente qu'une partie des mesures nécessaires à la maîtrise qualitative des projets dans l'organisation.

Comme nous l'avons déjà précisé, d'autres méthodologies doivent également être implémentées au sein de notre organisation : méthodologie de gestion de projet (prince 2), méthodologie de qualité de codage informatique (ITIL) et mise en place des PMO.

C'est l'ensemble de toutes ces mesures qui assurera l'évolution qualitative de l'organisation, et ce n'est qu'une fois leur implémentation effective qu'une évaluation objective pourra être menée.



## 8. Evolution de la méthodologie

Selon les prévisions et plannings établis par le projet SupDev, la méthodologie FUP arrive dans sa phase de stabilisation et d'acceptation par le S.P.F. Finances. Le périmètre du projet prévoyait la mise en place de la méthodologie et la prise en charge de la formation à celle-ci et aux outils logiciels qui la soutiennent. Un fois ces formations assurées, le projet SupDev prendra fin.

Au travers du Core Team constitué à l'avenir de spécialistes méthodologiques internes au S.P.F., la méthodologie évoluera en fonction des besoins des projets et de l'organisation. Les réflexions de ce Core Team laissent prévoir les évolutions suivantes :

- ↳ Adaptation à la spécificité des projets ;
- ↳ Adaptation à l'implémentation de la gestion de projet ;
- ↳ Adaptation à la maturité de l'organisation.

### 8.1. Adaptation aux spécificités des projets

Lors de l'Assessment nous avons déterminé l'ensemble des types de projets différents en cours au sein du S.P.F. Finances. Ces projets étaient classifiés comme suit :

- ↳ les projets internes s'appliquant souvent aux systèmes anciens (soit les projets « legacy »),
- ↳ les projets internes bureautiques ou informatiques
- ↳ les projets externes utilisant souvent les nouvelles technologies.

Actuellement FUP couvre de manière optimale les projets de type 2 et de type 3 a forte connotation « métier ».

Mais qu'en est-il des projets uniquement ICT, sans aspect « métier » ou à aspect « métier » très limité comme les projets d'achats de hardwares ou de softwares commerciaux, les projets de calcul d'impôt (basculement du « legacy » aux nouvelles technologies), ....

L'adaptation de la méthodologie à ces projets est prévue, avec pour objectif principal de centraliser les informations dans les outils de modélisation afin d'en assurer un meilleur suivi et une compréhension commune à tous.

Cette adaptation se fera par la définition des disciplines impliquées et la réduction ou l'extension des exigences de celles-ci en fonction du type de projet.

Un autre niveau d'adaptation est celui de la prise en compte de la maintenance adaptative des applications déjà en production (de type « legacy » ou de type « nouvelles technologies »).

La volonté affichée est que toute adaptation de telles applications soit un projet bien défini qui serait soumis aux mêmes règles méthodologiques qu'un projet normal de développement, tant interne qu'externalisé.

L'adaptation méthodologique sera moindre dans le sens où c'est au niveau des itérations que l'importance du projet sera déterminée.

## 8.2. Adaptation à la maturité de l'organisation

L'idée principale est d'imposer de manière différenciée les standards méthodologiques en fonction de la maturité acquise par les différents intervenants à un projet. Ainsi plus les concepts méthodologiques seraient acquis, plus l'organisation pourrait demander à ses acteurs d'approfondir les développements en utilisant toutes les possibilités de la méthode, enrichissant ainsi le travail fourni.

Pour ce faire, FUP pourrait être adapté de trois manières différentes.

- ↳ Light
- ↳ Médium
- ↳ Full

La méthode light reprendrait tous les concepts et livrables obligatoires dans le développement d'un projet mais allègerait les procédures de travail afin de permettre d'axer le travail plus sur le fond que sur la forme.

La méthode médium serait l'approfondissement de la méthode light en ce sens que tous les concepts obligatoires devraient être maîtrisés parfaitement. Des concepts supplémentaires seraient ajoutés au fur et à mesure et rendus progressivement obligatoires de manière à en acquérir la connaissance et la manipulation.

La version full serait l'approfondissement de la méthode médium dans laquelle tous les concepts seraient rendus obligatoires.

Ces trois niveaux dans la méthodologie seraient accompagnés d'un contrôle de qualité permanent qui permettraient de confirmer ou d'infirmer le niveau atteint par le projet et permettrait une adaptation corrective de celui-ci.

Le but final de ce nivellement méthodologique est d'amener l'ensemble de l'organisation à un niveau de connaissance et de maîtrise méthodologique élevé.



### **8.3. Adaptation à l'implémentation de la gestion de projet**

Ainsi que nous l'avons déjà exposé, l'implémentation d'une méthodologie de développement de projet ne couvre qu'une partie de son organisation. La notion de gestion de projet est aussi fort importante et indissociable de celle de développement.

Les méthodologies associées à cette gestion sont aussi nombreuses et variées. Au travers du projet « Gestion de projet » le S.P.F. Finances a pris la décision d'implémenter la méthodologie Prince 2, dont les principes sont basés sur l'itération et la notion de GO/NOGO à la clôture de chaque étape. L'implémentation de cette méthodologie est prévue pour début 2007.

L'intégration de la méthodologie de gestion de projet avec celle de développement est donc facilitée par la manipulation de concepts identiques tels la notion d'itération et l'intégration de fonction et profils métiers similaires.

Dans le cadre de FUP, et principalement au niveau de la fonction de chef de projet, il faudra intégrer les principes initiés par Prince 2 en favorisant les références aux tâches de gestion au travers des documents de discipline et des modes opératoires. Cette intégration devra aussi être prise en compte par « Gestion de projet » afin de permettre un référencement bidirectionnel aux deux méthodologies.

L'intégration des concepts et des tâches à accomplir par le chef de projet sera aussi grandement facilitée par la mise en œuvre de la suite logicielle de gestion de projet choisi par l'Administration. Les deux suites logicielles pourront s'interfacier entre elles de manière à apparaître comme une application unique permettant d'avoir une vue générale et centralisée du projet.

Cette centralisation des informations offre aussi un autre avantage : elle pourra être utilisée également par les PMO's comme outil de contrôle et de consolidation entre projets offrant ainsi aux managers du S.P.F. Finances un outil d'audit et de reporting.

## **9. Réutilisabilité de la méthodologie FUP dans le cadre d'autres organisations**

La méthodologie FUP a été établie pour être implémentée dans le cadre du S.P.F. Finances, en tenant compte de ses spécificités, de ses besoins et de ses contraintes.

Cette méthodologie se veut directive et contraignante car elle impose une marche à suivre stricte, imposant au fur et à mesure les productions attendues et les livrables formatés qui en découlent.

Ce désir de guidance à travers les disciplines, les modes opératoires et les livrables préétablis répond ainsi à la volonté fortement affichée par l'organisation de standardiser ses productions en documentant de manière linéaire et complète ses développements.

En effet notre organisation est imposante de par sa taille et de par le nombre de développements pris en charge suite à la réforme CoperFin. La gestion du développement de tous ces projets requerrait donc une méthodologie directive.

Les notions de réutilisabilité des codes produits et d'uniformisation des concepts ont été mises en place non seulement pour diminuer les coûts de production mais aussi pour assurer la facilité de maintenance et la compréhension générale des applications mises au point. Le glossaire transversal, le dictionnaire des données et les conventions de nommage en sont les exemples les plus significatifs.

### *FUP et l'organisation*

La méthodologie FUP s'adresse avant tout à une organisation importante et structurée. Cette structure doit disposer de ressources stables spécialement dédiées au développement de projets. Cette stabilité est requise afin d'assurer au mieux la formation à la méthodologie. L'ancrage des concepts et la maîtrise des techniques est en effet primordiale pour assurer le bon déroulement du processus de développement.

Pour toutes ces raisons, le déploiement de notre méthodologie dans de petites structures n'est pas recommandable car elle alourdirait inutilement le processus de développement. Les intervenants aux projets devraient cumuler les fonctions et le risque est très fort qu'ils passent plus de temps sur le formalisme du développement que sur le fond du projet.

L'organisation qui adopterait FUP comme méthodologie de développement doit être consciente de son aspect directif. Elle devra en effet mettre en place une structure de suivi et de guidance des projets. Les comités de pilotage et les comités de direction sont en effet au cœur du processus décisionnel.

Enfin, la méthodologie FUP ne peut se suffire à elle-même dans le processus de gestion d'un projet. Elle demande que l'organisation ait adopté, de préférence préalablement, une méthodologie de gestion de projets.



### FUP et les projets

Dans l'état actuel de développement de FUP seuls deux types de projets sont envisagés : les projets qui ont recours à l'aide de sociétés extérieures et qui suivent le trajet classique de l'appel d'offre suite à une préétude, et les projets purement internes. La caractéristique de ces projets est qu'ils ont une forte dimension « métier » et concernent la mise en place d'une application informatique.

Au niveau de la complexité des projets, FUP ne fait pas encore de distinction formelle entre les différents niveaux qui peuvent être rencontrés ; ce qui peut entraîner une lourdeur assez forte pour les projets de faible complexité ou de petite taille.

Les évolutions futures de la méthodologie prendront en compte ces aspects de complexité.

### FUP est inspiré de UP

Notre méthodologie s'est fortement inspirée des préceptes préconisés par UP (Unified Process) et de son pendant commercial RUP.

UP est souvent assimilée aux méthodologies de développement AGILES. Cependant la maîtrise des détails des processus de travail et le formalisme demandé pour ses productions ont tendance à obscurcir la plupart des caractéristiques AGILES qui sont incorporées dans sa philosophie.

La méthodologie FUP dans sa conception possède donc les qualités mais aussi les défauts d'UP. Cependant lorsque nous confrontons ses défauts et qualités aux exigences organisationnelles propres au S.P.F. Finances, nous constatons que ce qui était considéré comme défaut s'avère en fait un bienfait. En effet, le formalisme du résultat est d'importance égale à la qualité résultat. En d'autres termes, le fond et la forme.

Ce souci de formalisme n'est pas nécessairement une exigence fondamentale pour toute organisation qui souhaiterait adopter la méthodologie FUP.

### FUP et les outils de modélisation

Bien que nous ayons développé des modes opératoires précis, faisant le lien entre la méthodologie et les outils de développement, FUP est indépendant de toute suite logicielle. Ce n'est pas l'outil qui guide la méthodologie.

Cette indépendance méthode/outils permet l'adoption de FUP sans contrainte concernant la présence ou non d'outils de modélisation et de développement.

Cependant, en tenant compte de la philosophie de FUP et plus particulièrement des aspects de standardisation et de formalisation qu'elle préconise, le recours à une suite logicielle est fortement recommandé.

## 10. Conclusion

Dans la présente étude, nous avons relaté les étapes successives d'un projet d'implémentation d'une méthodologie de développement au sein d'un Service Public Fédéral Finances en pleine réforme organisationnelle et fonctionnelle. Nous nous sommes autant attardé sur la méthode de construction de la méthodologie que sur les fondements de celle-ci.

L'analyse détaillée du contexte nous a permis de bien comprendre les enjeux et les contraintes assignées au projet « Support au Développement ».

L'exposé de la démarche entreprise pour la réalisation de ce projet a voulu démontrer l'approche rationnelle et pragmatique adoptée pour mener à bien le projet. Nous avons tenu compte de la volonté managériale de standardiser les processus de travail en mettant en place un glossaire transversal à toute l'organisation ainsi qu'un dictionnaire des données centralisateur soutenu par des conventions de nommage bien déterminées.

Les enseignements retirés des projets pilotes à travers lesquels nous avons testé la méthodologie développée nous ont permis d'établir une liste de recommandations pour l'implémentation à grande échelle. Nous avons exposé les mesures correctives à adopter et les évolutions à prévoir dans un futur proche.

Ensuite, nous nous sommes intéressé à l'aspect réutilisable de la méthodologie au sein d'autres organisations, comparables ou non au S.P.F. Finances et nous avons conclu que, moyennant le respect de critères tels que la taille de l'organisation elle était transposable.

Enfin, nous ne pourrions terminer ce travail sans porter un regard critique sur la manière dont a été mené ce projet. Nous avons pu constater que l'Assessment mené au sein de l'organisation s'est trop vite contenté d'une approche superficielle du mode de fonctionnement des services informatiques du S.P.F. Finances. De manière générale, nous avons constaté que la phase d'analyse préliminaire du projet n'a pas été assez approfondie et que la maîtrise du contexte n'était pas optimale.

En ce qui concerne la gestion du projet à proprement parler, nous avons regretté le manque de synchronisation des implémentations méthodologiques prévues. En effet le S.P.F. Finances met actuellement en place deux projets comparables au nôtre afin d'implémenter une méthodologie de gestion de projet et une méthodologie de gestion managériale des programmes d'investissement. Ces deux projets d'implémentation utiliseront pour leur développement la méthodologie FUP que nous avons mise au point et il sera intéressant de pouvoir en évaluer la pertinence et l'adéquation. En effet, il s'agit ici de projets sans dimension ICT importante.

Une autre piste de réflexion serait aussi celle de l'analyse préliminaire concernant l'intégration possible des philosophies de ces diverses méthodologies (management, gestion et développement) au sein de notre organisation.



## 11. Bibliographie

Remarque préliminaire :

Les documents de référence qui ont été utilisés pour la rédaction de ce mémoire sont essentiellement issus de la documentation interne au SPF Finances.

### **Concernant les réformes Copernic et CoperFin :**

Documentations issue des rapports finaux des groupes de travail BPR :

- *livre des processus*
- *livre des fonctions*
- *livre organisation 1*
- *livre organisation 2*
- *livre réalisation 1*
- *livre réalisation 2*

Note au Gouvernement, "*Fondements de la modernisation de l'administration fédérale*", 2000

Service public fédéral Finances, *Rapport annuel 2003*, octobre 2005

Service public fédéral Finances, *Plan d'appui ICT 2006*, 15 janvier 2006

Service public fédéral Fiances, *Le rapport final approuvé "Elaboration conceptuelle d'un nouvel organigramme pour le SPF Finances"*, (décembre 2000)

Service public fédéral Personnel et Organisation, *Copernicus – Au centre de l'avenir – La réforme de l'Administration fédérale belge*, octobre 2002

### **Concernant le projet SupDev (Support au Développement)<sup>1</sup>**

Documentation issue de la gestion et des livrables du projet SupDev :

- *rapport de la commission d'évaluation des offres, octobre 2004<sup>2</sup>*
- *réponse à l'appel d'offre de la société XXX ;*
- *rapports du Comité de Pilotage du projet SupDev (un par mois depuis janvier 2005*
- *notes internes du groupes d'étude « Core Team »*
- *rapports d'avancements des projets pilotes*
- *notes de synthèse adressées au comité de direction*
- *livrables de SupDev :*
  - *FUP-Introduction v0 [1].9.doc*
  - *FUP-Cycle de Vie v1 [1].4.doc*
  - *FUP-Discipline Modélisation Métier v0 [1].8.doc<sup>3</sup>*
  - *FUP-Discipline Exigences v0 [1].9.doc*

---

<sup>1</sup> Pour tous ces documents nous avons été partie prenante à leur construction et leur rédaction en tant que Chef de Projet métier, membre du Comité de Pilotage et membre du « Core Team »

<sup>2</sup> Repris en annexes

<sup>3</sup> Repris en annexes

- *FUP-Discipline Analyse Conception v0 [1].11.doc*
- *FUP-Discipline Implémentation v07.doc*
- *FUP-Stratégie de test maitre v0 [1].3.doc*
- *FUP-MO-Discipline Modélisation Métier Fr v0 [1].1.doc<sup>4</sup>*
- *FUP-MO-Discipline Tests Fr v0 [1].1.doc*
- *FUP-MO-Discipline Exigences Fr v0 [1].1.doc*
- *FUP-MO-Discipline Analyse Fr v0 [1].1.doc*
- *FUP-MO-D1-Acteurs Métier Fr v0 [1].1.doc*
- *FUP-MO-D1-Domaine Métier Fr v0 [1].1.doc*
- *FUP-MO-D1-Glossaire Métier Fr v0 [1].1.doc*
- *FUP-MO-D1-Processus Métier Fr v0 [1].1.doc<sup>5</sup>*
- *FUP-MO-D1-Règles Business Fr v0[1].1.doc*
- *FUP-MO-D2-Acteurs Système Fr v0 [1].1.doc*
- *FUP-MO-D2-Architecture Fonctionnelle Fr v0 [1].1.doc*
- *FUP-MO-D2-Exigences Fonctionnelles Fr v0 [1].1.doc*
- *FUP-MO-D2-Exigences Supplémentaires Fr v0 [1].1.doc*
- *FUP-MO-D3-Architecture Logicielle Fr v0 [1].1.doc*
- *FUP-MO-D3-Interface Utilisateur Fr v0 [1].1.doc*
- *FUP-MO-D3-Modèle des données Fr v0 [1].1.doc*
- *FUP-MO-D4-Tests intégration Fr v0 [1].1.doc*
- *Convention\_de\_nommage-2[1].52.doc<sup>6</sup>*

Service public fédéral Finances, Service d'encadrement ICT, *Cahier des charges « Support au Développement*, août 2004

### **Concernant l'analyse méthodologique et les diverses référence à UP**

Boehm B, Turner R, *Balancing Agility and Discipline : a guide for the Perplexed*, Addison-Wesley, Pearson Education, ISBN 0-321-18612-5

Jacobson I, Booch G, and Rumbaugh, *Le Processus unifié de développement logiciel*. c/o Eyrolles, 2000, ISBN 2-212-09142-7

Kruchten Philippe, *Introduction au Rational Unified Process*, c/o Eyrolles, 2000, ISBN 2-212-09104-4

Jacobson I, Booch G, and Rumbaugh, *Le guide de l'utilisateur UML*, c/o Eyrolles, 2000 ISBN 2-212-09103-6

Gullion Tom, *Unified Process using the Borland® ALM Solution*, Borland Software Corporation <http://bdn.borland.com/article/images/31950/BorlandALM-UnifiedProcess.pdf>

Robert M, *Agile Software Development: Principles, Patterns and Practices*, Prentice Hall PTR. 2003, Upper Saddle River, NJ.

---

<sup>4</sup> Repris en annexes

<sup>5</sup> Repris en annexe

<sup>6</sup> Repris en annexes



## 12. Annexes

### 12.1. Rapport Evaluation Support Au Développement.

Rapport rédigé par le chef de projet SupDev.

Pour des raisons de confidentialité, le nom des sociétés soumissionnaires a été modifié en A, B, C et D et les évaluations portant sur les prix enlevées.

#### Analyse des critères d'évaluation :

Le tableau ci-dessous reprend les moyennes des 4 évaluateurs. Chaque critère est pondéré selon les poids indiqués dans le cahier des charges.

Chaque critère et sous-critère sera commenté individuellement. Les cotations sont le résultat de l'évaluation des membres de la commission, les avis de l'expert de la commission ont été pris en compte dans cette cotation, qui a pu être revue, selon les cas, suite aux démonstrations proposées par les firmes figurant dans la *short-list*.

	Poids	A	B	C	D
<b>B. Qualité de la solution logicielle</b>	<b>30</b>	21,9975	21,9975	20,225	25,8375
Fonctionnalités de gestion des spécifications	4	3,1875	3,1875	2,625	3,5375
Analyse Conceptuelle	6	4,25	4,25	3,825	4,625
Fonctionnalités « Entreprise Data Modelling »	5	3,225	3,225	4,3625	4,3625
Fonctionnalités Design – IDE- Java 2EE	6	4,2675	4,2675	3,9125	5,5
Fonctionnalités « Outils de Test »	4	3,13	3,13	2,375	3,75
Intégration des différents composants logiciels de l'offre, reprise de l'existant et environnement multiutilisateurs	5	3,9375	3,9375	3,125	4,0625
<b>C. Qualité de la méthodologie</b>	<b>15</b>	9,4375	12,495	11,4025	12,56
Complétude méthodologie et complétude livrables proposés	8	5,3	6,815	6,355	6,925
Adaptation environnement S.P.F.	7	4,1375	5,68	5,0475	5,635
<b>D. Intégration Méthode – Outils</b>	<b>5</b>	3,875	4,15	3,375	4,1625
<b>E. Lisibilité et conformité de l'offre (5 %)</b>	<b>5</b>	3,4375	4,275	3,875	4,275
<b>F. Case Study (5 %)</b>	<b>5</b>	0,75	3,9375	3,125	4,5
<b>G. Formation – Documentation (3 %)</b>	<b>3</b>	1,175	2,35	2,135	2,1375
<b>H.SLA – Support (2 %)-</b>	<b>2</b>	0,9625	1,475	1,0825	1,15
<b>TOTAL</b>	<b>65</b>	41,635	50,68	45,22	54,6225

#### Critère 2 : Qualité de la solution logicielle proposée (30 %)

##### Fonctionnalités de gestion des spécifications (4 %)

La solution proposée par D (Caliber RM) couvre toutes les spécifications demandées. Elle offre la possibilité aux utilisateurs Business et ICT de capturer, de structurer et de stocker les spécifications des applications au moyen d'une interface intuitive et pouvant être facilement adaptée aux besoins



spécifiques du S.P.F. Finances. Les catégories de spécifications proposées (fonctionnelles, techniques,...), ainsi que leurs propriétés peuvent être complétées et adaptées en fonction des besoins spécifiques du S.P.F. Pour saisir les spécifications, l'utilisateur dispose des écrans fournis par l'outil, ceux-ci pouvant être adaptés en fonction des besoins du S.P.F. Une autre possibilité consiste en le marquage de portions de documents MS-Word. De plus, les spécifications peuvent être importées automatiquement dans CaliberRM, si elles sont disponibles dans un format MS-WORD ou XML et ce à partir d'un style ou d'une structure à définir. Ceci a pour avantage d'automatiser une grande partie de la saisie des spécifications dans l'outil, tout en permettant au S.P.F. de définir librement le format utilisé par les utilisateurs business et ICT pour les saisir. Ceci permet également de garantir une meilleure adéquation entre la méthodologie et les outils utilisés. Il permet également le lien entre spécifications et des documents de tous types pouvant les compléter. Le partage des informations fournies vers les utilisateurs ICT est assuré au moyen de l'intégration avec le produit StarTeam décrit ci-dessous. De plus, le logiciel client est intégré directement à partir des différents outils proposés dans l'offre D, à l'exception d'E/R Studio. L'application est également accessible au moyen d'un browser Web, ce qui permet son utilisation sans installation physique du logiciel sur les postes des utilisateurs. Les facilités de reporting sont étendues, simples à mettre en œuvre et à paramétrer. En résumé, Caliber RM convainc grâce à sa facilité d'utilisation et son aspect intuitif pour les utilisateurs Business du S.P.F., ainsi que sa flexibilité quant à sa mise en œuvre dans l'environnement spécifique du S.P.F. Finances.

RequisitePro, la solution proposée par B et A couvre également les spécifications demandées, en offrant la possibilité de saisir directement les besoins à partir de l'interface utilisateur de l'outil ou au moyen de macros Word permettant à l'utilisateur de marquer et de d'introduire des portions de documents dans la base de données des spécifications. Cependant, les possibilités d'importations automatisées de spécifications contenues dans des documents Word ou d'autres formats sont plus limitées, plus particulièrement dans la gestion des styles de documents et de documents XML, et sont moins efficaces dans le traitement automatique de documents fournis par les utilisateurs. RequisitePro met l'accent sur la saisie interactive des spécifications, qui n'est pas toujours compatible avec le traitement d'un grand volume de celles-ci. Comme dans CaliberRM, les catégories de spécifications proposées (fonctionnelles, techniques,...), ainsi que leurs propriétés peuvent être complétées et adaptées en fonction des besoins spécifiques du S.P.F. La traçabilité des spécifications vis-à-vis d'autres résultats est assurée par l'utilisation d'une base de données DB2, compatible avec les standards du S.P.F. Finances. RequisitePro est également accessible au moyen d'un browser Web, ce qui permet son utilisation sans installation physique du logiciel sur les postes des utilisateurs. En résumé, si RequisitePro fournit un ensemble de fonctionnalités étendues, sa plus grande complexité de mise en œuvre, son couplage fort avec Word par l'utilisation de macros spécifiques et ses possibilités moindres dans le traitement automatique de documents structurés le rendent moins indiqué dans l'environnement spécifique du S.P.F. Finances que CaliberRM.

Power Designer, la solution proposée offre deux chemins pour saisir les spécifications : le module BPM (business process Modelling) et le module OOM (Object Oriented Model). Le module BPM modélise les spécifications de haut niveau en utilisant un formalisme graphique qui lui est propre et qui devra être maîtrisé par les utilisateurs Business, ce qui risque de freiner son acceptation par ceux-ci. Le module OOM permet la saisie des spécifications à partir de l'interface utilisateur de Power Designer qui est moins flexible et moins intuitive que celle offerte par CaliberRM et RequisitePro. Des possibilités d'importation automatique de document Word sont également fournies. Les informations fournies dans les deux modules sont stockées et consolidées dans le repository de Power Designer qui assure la traçabilité vers les autres étapes du développement. Les possibilités de reporting sont étendues et flexibles. La solution proposée n'est pas utilisable à partir d'un browser Web, ce qui implique l'installation physique du logiciel sur tous les postes clients. En résumé, si la solution proposée par C offre les fonctionnalités minimales exigées, elle est moins riche fonctionnellement et moins flexible que celle des autres soumissionnaires. De plus, imposer l'emploi d'un formalisme spécifique BPM aux utilisateurs Business risque de retarder l'acceptation



de l'outil par ceux-ci et de créer une confusion avec le formalisme UML qu'ils devront utiliser dans la phase d'analyse.

### **Analyse Conceptuelle (6 %)**

Together Control Center, la solution proposée par D, offre toutes les fonctionnalités demandées. Il permet la réalisation de modèles définis dans le standard UML 1.4 au moyen d'une interface utilisateur intuitive et qui permet un démarrage rapide des utilisateurs. La traçabilité ainsi que la navigation entre diagrammes est gérée de manière intuitive au moyen d'hyperliens et de touches de raccourcis entre ceux-ci. L'outil laisse à l'utilisateur une grande flexibilité dans l'organisation de son travail, ce qui facilitera l'acceptation de l'outil par les utilisateurs. Ceci a cependant comme inconvénient d'imposer des procédures de contrôle de qualité strictes afin de garantir le respect des procédures méthodologiques du S.P.F. Le lien avec les spécifications business est assuré par une intégration du client CaliberRM avec Together. Les fonctionnalités de reporting fournies sont étendues et facilement paramétrables, les modèles réalisés pouvant être publiés en format Html. Together permet l'interaction bidirectionnelle entre modèles UML et les sources Java qui y correspondent au cours de la réalisation de ceux-ci. De plus, Together permet le reverse engineering à posteriori de code Java existant vers les modèles UML correspondants. Together permet également la définition et l'utilisation de métriques et d'indicateurs de qualité qui permettent de suivre de manière synthétique l'évolution d'un projet. En résumé, Together Control Center combine les fonctionnalités demandées avec une interface utilisateur intuitive et flexible, ce qui est un atout important dans le contexte de l'introduction de ce type d'outils au S.P.F. Un point d'attention important lors de l'implémentation de l'outil sera la mise en place de procédures de contrôle de qualité permettant aux utilisateurs de rester « en phase » avec la méthodologie du S.P.F.

XDE Modeler, la solution proposée par B et A, offre toutes les fonctionnalités demandées. Il permet la réalisation de modèles UML au standard UML 1.4, assure la traçabilité avec les spécifications définies dans RequisitePro et ce d'une manière transparente pour l'utilisateur. Les possibilités de reporting sont étendues en utilisant Rational SODA et peuvent être publiés en format HTML. Cependant, XDE Modeler est handicapé par une interface utilisateur lourde, complexe et peu intuitive qui, dans le contexte spécifique du S.P.F., risque de freiner son acceptation par les analystes business et ICT du S.P.F. En résumé, XDE Modeler est un outil complet mais complexe à utiliser et à mettre en œuvre dans le contexte spécifique du S.P.F.

Power Designer OOM, la solution proposée par C, offre toutes les fonctionnalités demandées. Il permet la réalisation de modèles UML au standard UML 1.4, assure la traçabilité avec les spécifications définies dans les phases précédentes du développement et stockées dans le repository de Power Designer et ce d'une manière transparente pour l'utilisateur. Les possibilités de reporting sont étendues en utilisant les fonctionnalités de reporting incluses dans OOM et peuvent être publiés en format HTML. En résumé, Power Designer OOM offre les fonctionnalités demandées mais en étant moins riche fonctionnellement que ses concurrents.

### **Fonctionnalités « Entreprise Data Modelling » (5 %)**

D propose E/R Studio ainsi que le logiciel E/R repository pour couvrir ce besoin. Le produit répond aux fonctionnalités demandées : les possibilités de modélisation relationnelles au niveau conceptuel, logique et physique sont présentes et accessibles au moyen d'une interface utilisateur intuitive qui met en œuvre des concepts universels dans le domaine de la modélisation de données, ce qui facilitera son introduction auprès des administrateurs de données du S.P.F. De plus, l'outil permet la gestion précise de tous les paramètres nécessaires à la gestion et à la génération de bases de données dans l'environnement technique du S.P.F., ainsi que les transformations complexes entre modèle conceptuel et modèle physique. Il permet également, moyennant le développement de scripts VB, la documentation via une connexion ODBC, de schéma de bases de données et fichiers non relationnels, ce qui permettra d'intégrer dans les modèles les informations disponibles dans les



systèmes « legacy » du S.P.F. Les possibilités de conversion des modèles E/R vers les diagrammes de classe sont assurées via des fonctionnalités d'import/export en format XMI. La prise en compte de modèles multidimensionnels le rend apte à la modélisation de data warehouses. On peut regretter que l'intégration du repository E/R Studio dans les référentiels StarTeam ne soit pas possible sans développement d'une interface spécifique. En résumé, E/R studio et les produits associés fournissent une solution complète, flexible à mettre en œuvre et s'intégrant d'une manière satisfaisante aux autres composants de l'offre, moyennant développement de l'interface mentionnée ci-dessus.

B et A proposent les fonctionnalités DataModeller incluses dans XDE Modeler. La description de ces fonctionnalités est quasi-absente des offres mentionnées. XDE Modeler utilise des diagrammes UML mettant en œuvre des stéréotypes spécifiques à la modélisation relationnelle, logique et physique. Cette solution a pour avantage d'intégrer les diagrammes de modélisation de données aux autres diagrammes UML réalisés au cours des phases d'analyse et de design. Cependant, elle a pour inconvénient d'être moins riche fonctionnellement et de ne pas tenir compte des spécificités du métier d'administrateur de données dans un environnement fonctionnel et technique complexe. En résumé, XDE Modeler fournit les fonctionnalités minimales demandées, très bien intégrée avec les autres composantes de l'offre. Cependant, le choix de l'utilisation de concepts peu familiers dans un environnement d'administration de données le rend peu adapté à l'environnement organisationnel et technique du S.P.F.

C propose avec Power Designer CDM et PDM un produit qui répond aux fonctionnalités demandées : les possibilités de modélisation relationnelles au niveau conceptuel, logique et physique sont présentes et accessibles au moyen d'une interface utilisateur intuitive qui met en œuvre des concepts universels dans le domaine de la modélisation de données, ce qui facilitera son introduction auprès des administrateurs de données du S.P.F.. L'intégration des modèles de données dans le même repository que les modèles UML permet une conversion simple et directe vers les diagrammes de classe correspondants. L'outil permet de prendre en compte toutes les spécificités de l'environnement DB du S.P.F. ainsi que les transformations complexes entre modèle conceptuel et modèle physique. Il permet également, moyennant le développement de scripts VB, la documentation via une connexion ODBC, de schéma de bases de données et fichiers non relationnels, ce qui permettra d'intégrer dans les modèles les informations disponibles dans les systèmes « legacy » du S.P.F. La prise en compte de modèles multidimensionnels le rend apte à la modélisation de data warehouses. En résumé, les modules de Power designer proposés fournissent une solution complète, flexible à mettre en œuvre et s'intégrant d'une manière transparente aux autres composants de l'offre C.

### **Fonctionnalités Design – IDE- Java 2EE (6 %)**

Together Enterprise Studio, proposé par D, intègre les fonctionnalités de design UML présentes dans Together avec les fonctionnalités de l'environnement de programmation J2EE JBuilder Enterprise Edition. Les fonctionnalités de design UML proposées sont complètes et relativement aisées à mettre en œuvre. Il permet au développeur d'intégrer facilement dans le design des applications les informations relatives aux bonnes pratiques (design patterns) et API les plus courantes dans le monde J2EE (Struts, Xerxes, Xalan,...) et de pouvoir traduire quasi instantanément ces informations de design en code J2EE . Il permet également la modélisation de messages XML. Ces fonctionnalités complexes sont pilotées par des Wizard permettant une mise en œuvre efficace. L'environnement de design et développement permet également la définition aisée des différents types de Bean pouvant être mis en œuvre dans un environnement J2EE Weblogic, ainsi que ses spécificités y compris la définition des descripteurs de déploiement. Tout ceci, combiné à l'intégration de CaliberRM, permet au développeur de concentrer ses efforts sur le développement du code lié à la concrétisation des spécifications fonctionnelles de l'application. Les fonctionnalités de reporting sont identiques à celles de Together Control Center. L'interface de l'IDE Java est, malgré sa complexité fonctionnelle, simple à appréhender et offre toutes les



fonctionnalités attendues en terme de support au développeur : analyse syntaxique du code, aide au remplissage,... Les fonctionnalités de build d'applications complexes sont supportées par ANT, standard de facto du marché. L'intégration avec le serveur d'application Weblogic est complète et transparente, y compris au niveau du debugging d'applications distribuées. En résumé, Enterprise Studio est une solution mûre et éprouvée qui offre un ensemble complet de fonctionnalités permettant d'augmenter simultanément la productivité du développeur ainsi que la qualité des applications développées et qui s'intègre parfaitement à l'environnement ICT du S.P.F.

XDE Développer, proposé par B et A, combine un environnement de modélisation UML, proche de XDE Modeler et un environnement de développement J2EE basé sur Eclipse. Les fonctionnalités de design UML sont complètes et permettent de réaliser les diagrammes UML 1.4. Cependant, comme dans XDE Modeler, l'interface utilisateur est lourde à mettre en œuvre et complexe. Les fonctionnalités de reporting sont identiques à celles de XDE Modeler. Il permet au développeur d'intégrer les informations relatives aux bonnes pratiques (design patterns), ainsi que de définir les différents types de beans de l'architecture J2EE. L'IDE Java intégrée à XDE Developer est Eclipse, dans sa version Java Développement Toolkit, enrichie par son intégration avec XDE Developer. L'environnement Eclipse, s'il évolue rapidement, n'a pas encore atteint le degré de maturité de JBuilder en termes de richesse fonctionnelle et de stabilité. Il peut être enrichi au moyen de plug-ins supplémentaires, payants ou open-source. Cependant l'intégration correcte de ces différents plug-ins, ainsi que la stabilité technique et l'évolutivité de la plate-forme résultante n'est pas aussi sûrement garantie. La construction d'applications complexe est réalisée au moyen d'ANT, standard de facto du marché. L'intégration avec l'environnement Weblogic est bonne en utilisant le plug-in Weblogic fourni par B. En résumé, si l'environnement de design UML proposé est de bonne qualité, mais handicapé par une interface utilisateur le rendant plus complexe à mettre en œuvre, l'environnement de développement Java proposé est moins riche fonctionnellement et moins abouti que JBuilder.

C propose la combinaison de Power designer OOM pour le design UML et d'Eclipse comme environnement de développement Java. Power Designer OOM permet de réaliser les diagrammes de design UML 1.4. Cependant, OOM apporte peu de valeur ajoutée quant au support des API J2EE les plus courantes et ne permet pas au développeur d'intégrer les composants relatifs à ceux-ci dans ses modèles. Il ne permet pas l'intégration de composants relatifs aux bonnes pratiques (design patterns). L'environnement Eclipse proposé est peu enrichi, même s'il implémente l'interaction bidirectionnelle entre modèles OOM et code Java.

Il devra être enrichi au moyen de plug-ins supplémentaires, payants ou open-source pour arriver à un niveau de fonctionnalité approchant celui de JBuilder mais sans en atteindre la stabilité technique. De plus, l'évolutivité de la plate-forme résultante n'est pas garantie. La construction d'applications complexe est réalisée au moyen d'ANT, standard de facto du marché. L'intégration avec Bea Weblogic est satisfaisante. En résumé, l'offre de C propose un ensemble de fonctionnalités minimales mais beaucoup moins susceptible d'augmenter la productivité de développeurs et la qualité des applications fournies

#### **Fonctionnalités « Outils de Test » (4%)**

D propose la suite Mercury, se composant de Test Director, QuicktestProfessional, Application Delivery Dashboard, Loadrunner et Borland Optimizelt. Cet ensemble d'outils permet de gérer et d'exécuter tous les types de test demandés, depuis les tests unitaires de composants jusqu'aux tests de performance, à partir d'une interface utilisateur intuitive.

TestDirector, l'outil permettant de manager les plans de test, est accessible via une interface Web, ce qui permet son utilisation par tous les acteurs concernés, sans installation de logiciel sur leur poste de travail. Il permet de combiner dans un même plan de test des tests manuels et automatisés (replay de scénarios créés par QuickTest Professional). L'intégration des différents composants de



la suite permet le transfert des informations pertinentes (plans de tests, résultats de tests, reporting) de manière aisée. La traçabilité entre les différents modèles réalisés au cours de l'analyse et du développement est assurée par une intégration poussée avec Starteam. Il est de plus configurable en fonction des choix méthodologiques et organisationnels du S.P.F.. On peut également noter que Borland Optimizelt permet l'analyse poussée du fonctionnement de composants sans interférer avec leur code exécutable, ce qui garantit une fiabilité totale des résultats de l'analyse.

La solution proposée est également éprouvée dans des environnements de développement et d'exploitation d'une complexité similaire à celui du S.P.F.. En résumé, les outils proposés répondent parfaitement aux besoins exprimés par le S.P.F., combinent richesse fonctionnelle, adaptabilité à la méthodologie choisie et à l'environnement spécifique du S.P.F., ainsi qu'une interface utilisateur permettant gérer efficacement les interactions entre les différents composants.

B et A proposent IBM Rational Test Manager, IBM Rational Functional Tester, IBM Rational Performance Tester, IBM Rational Purify Plus, IBM Rational XDE Tester. Ces outils permettent de gérer et d'exécuter tous les types de tests demandés, depuis les tests unitaires jusqu'aux tests de performance. Les différents composants de la suite sont intégrés et la traçabilité entre les différents modèles réalisés au cours de l'analyse et du développement est assurée par RequisitePro. Cependant, la mise en œuvre des logiciels proposés manque de flexibilité. En effet, Test Manager nécessite l'installation du logiciel sur les postes de travail des utilisateurs. En résumé, les outils proposés répondent aux besoins exprimés par le S.P.F., mais sont moins flexibles dans leur mise en œuvre que ceux proposés par D

C propose une solution basée sur le Framework de test open-source Hyades, intégrée à Eclipse. Les fonctionnalités proposées sont insuffisantes. En effet, si la solution proposée offre des fonctionnalités satisfaisantes en termes de tests unitaires et d'analyse à posteriori d'exécution d'applications et de capture de session, elle ne permet pas la gestion et la définition aisée de plans de tests fonctionnels et leur traçabilité avec le reste des modèles réalisés au cours de l'analyse et du développement. Ceci ne permet pas d'envisager son déploiement de la solution auprès des analystes business et donc ne permet d'implémenter une stratégie de tests couvrant tout le cycle de vie des applications. La solution proposée ne permet pas, telle que présentée, la gestion et l'exécution de tests de performance en tant que tels. Les possibilités de reporting sont limitées au strict minimum.

Si sur base de la documentation fournie dans l'offre, la solution proposée semblait répondre aux critères minimums exigés, les fonctionnalités démontrées au cours du case-study ont amené la commission à attribuer pour ce point une cote inférieure à la cote d'exclusion définie au point 1.2.13.3.1 du cahier des charges. En résumé, la solution proposée est un Framework qui, dans son état actuel, ne couvre pas les besoins du S.P.F.

#### **Intégration des différents composants logiciels de l'offre, reprise de l'existant et environnement multiutilisateurs (5 %)**

L'intégration des différentes composantes de l'offre D est basée sur l'outil de gestion de version et de configuration StarTeam dans lequel s'intègre Caliber RM. StarTeam permet la gestion et le partage des artefacts stockés dans son repository, à partir d'une interface Web ou de plug-ins vers les autres logiciels proposés dans l'offre. L'utilisation à partir d'une interface Web ne nécessite pas l'installation d'un client spécifique sur le poste de travail des utilisateurs. Il permet la définition aisée de projets, de Baseline à l'intérieur des projets. La collaboration et le partage d'objets inter-projets est également possible. Il permet la gestion des conflits entre différentes versions d'un même artefact. Des possibilités suffisantes de travail collaboratif sont présentes et permettent une gestion de workflow facilement paramétrisable entre les différents intervenants. Les droits d'accès aux différents objets peuvent être gérés à différents niveaux d'abstraction (projet, objet,...) et les informations relatives aux utilisateurs contenues gérées par Active Directory peuvent être utilisées. L'intégration des informations contenues dans CaliberRM est réalisée au moyen d'hyperliens. Ces liens permettent d'assurer la traçabilité dynamique entre les différents modèles et requirements



créées au cours du développement. Les modèles de données créés par E/R Studio et exportés en format XMI peuvent être placés sous le contrôle de StarTeam. Cependant le repository d'E/R Studio reste une composante indépendante de StarTeam. En contrepartie, le repository E/R Studio permet la constitution d'un dictionnaire de données relationnelles complexe utilisable par les administrateurs de données au niveau conceptuel et opérationnel. Il est important de noter que les concepts sous-jacents à StarTeam et aux outils associés sont des concepts maîtrisés par les utilisateurs du système actuel de gestion de version et de configuration du S.P.F. (PVCS), ce qui facilitera sa mise en œuvre et son bonne compréhension par les utilisateurs de tous profils. En ce qui concerne la reprise de l'existant, la migration des différents modèles et fichiers déjà réalisés avec les outils de la famille Rational est possible, mais demandera une intervention manuelle. En résumé, D propose une offre qui, si elle est moins riche fonctionnellement, sera plus facile à mettre en œuvre dans le contexte spécifique du S.P.F., car basée sur des concepts maîtrisés par les utilisateurs du S.P.F.

L'intégration des différentes composantes de l'offre B et A est assurée par la plate-forme UCM dont les composants principaux sont ClearCase et ClearQuest. Clearcase permet la gestion de versions et de configuration de tous les modèles et artefacts créés au cours du cycle de développement, ainsi que l'intégration avec les spécifications gérées par RequisitePro. Il permet le partage d'objets dans un environnement multiutilisateurs complexe pour des projets de grande taille en intégrant les différents objets à partager en unités logiques. Ce sont ces unités logiques qui sont gérées par l'utilisateur. A l'intérieur d'un projet, différents rôles peuvent être définis et en fonction de ces rôles, un environnement de travail distinct (stream) peut être défini qui décrit à quelles versions de quels objets l'utilisateur peut accéder. Cette approche permet une gestion très fine des accès et du partage des objets au prix d'une plus grande complexité dans la mise en œuvre et l'utilisation de l'outil. Le fait que les concepts utilisés par ClearCase soient plus complexes et différents de ceux utilisés par le système actuel de gestion de version et de configuration du S.P.F. (PVCS) rendra son acceptation par les utilisateurs et sa mise en place plus complexe.

Clearquest permet la gestion de workflow et des activités liés à la gestion des demandes de changement. Clearquest fournit un environnement complet très bien intégré à ClearCase. On regrettera que l'accès à Clearquest via une interface Web ne soit possible qu'au moyen d'un produit optionnel. En ce qui concerne la reprise de l'existant, B donne peu d'informations quant à la reprise de projets de développement déjà réalisés dans l'environnement JBuilder déjà utilisé par le S.P.F. dans différents projets. En résumé, l'offre d' B et A offre un ensemble puissant de fonctionnalités permettant de gérer le contenu de projets complexes. En contrepartie, elle est complexe à mettre en œuvre et est basée sur des concepts qui ne sont pas familiers dans le contexte du S.P.F.

L'intégration de la solution proposée par C doit prendre en compte la séparation totale entre les mécanismes de gestion des modèles et les mécanismes de gestion des sources des applications développées. La gestion des modèles est réalisée au moyen d'un repository stocké dans une base de données relationnelle et permet de ce fait de gérer de manière simple et efficace les accès concurrents, le versioning et le partage des modèles. En ce qui concerne les sources, C propose de gérer leur versioning et leur partage au moyen d'un plug-in PVCS, produit déjà utilisé par le S.P.F. Cette architecture pose un problème essentiel : le maintien de la synchronisation entre modèles et code, ainsi que la définition et le maintien de Baseline (ensemble de modèles et de sources constituant une tout logique). L'architecture proposée ne permet pas cette synchronisation « out of the box ». C propose de gérer cette synchronisation au moyen de programmes développés sur mesure par leurs soins. Cette solution n'est pas satisfaisante dans le contexte du développement d'applications complexes au sein du S.P.F., car elle n'offre pas les garanties de fiabilité et de sécurité nécessaires. En résumé, si examinées séparément, les solutions proposées par C sont bonnes, les modalités d'intégration des deux produits ne permettent pas de garantir et de vérifier la cohérence des différentes composantes constituant une application.



### **Critère 3 : Qualité de la méthodologie.**

#### **Complétude méthodologie et complétude livrables proposés (8 %)**

A propose d'implémenter RUP (Rational Unified Process). RUP est une méthodologie basée sur un modèle itératif, bien adapté au développement d'applications complexes. A a pu mettre implémenter cette méthodologie dans le cadre du projet Copernic du Ministère des Finances en France. Cependant, dans son offre, A se contente de décrire les grandes lignes de RUP et d'un nombre limité de livrables réalisés dans le contexte de Copernic. Ceci ne permet pas de juger correctement de l'approche stratégique d'A dans le contexte de l'implémentation de RUP. A propose la création d'un site intranet basé sur RUP permettant la publication de standards méthodologiques du S.P.F.

B propose également d'implémenter RUP pour couvrir tout le cycle de vie des applications. RUP est complet, complexe et peut-être paramétrisé en fonction des besoins spécifiques de l'organisation. La description des artefacts de RUP est peu détaillée. B propose la réalisation d'un site intranet basé sur RUP permettant la publication de standards méthodologiques du S.P.F.

C propose d'implémenter une variante d'Unified Process, pendant « Open source » de RUP. On peut regretter que l'offre ne contienne que très peu de précision quant aux livrables proposés par l'offre et ceux dont l'emploi est envisagé dans le cadre du projet. De plus, C ne prévoit pas la publication de son guide méthodologique sur un site Intranet, mais seulement en format Word.

D propose également RUP, mais en l'encadrant dans un Framework 3Dvue Business Blueprint. D fournit une bonne description des concepts mis en œuvre par RUP ainsi que des outils disponibles pour le gérer. D prévoit la réalisation d'un site intranet basé sur RUP permettant la publication de standards méthodologiques du S.P.F.

#### **Adaptation environnement S.P.F. (7%)**

A fournit un plan d'implémentation sommaire, peu détaillé et ne donnant qu'une vue très synthétique de sa démarche. De plus, A n'accorde que peu d'importance au change management qui accompagne impérativement ce type de projet dans le contexte du S.P.F. Ceci rend l'approche d'A globalement peu satisfaisante.

B fournit un plan d'approche extrêmement détaillé, accompagné d'une démarche rigoureuse et claire, mettant en œuvre un processus d'accompagnement qui fournit un support important aux équipes du S.P.F. impliquées dans le projet. Les étapes et rôles des différents acteurs, internes et externes, qui devront participer à la mise en œuvre de la solution est décrit avec une extrême précision. B a une vue claire des spécificités du S.P.F. et des difficultés liées à l'implémentation de nouvelles méthodes de travail et d'outils informatiques complexes dans une grande organisation.

C choisit une approche très interactive pour l'implémentation de sa méthodologie, en attendant du S.P.F. un input important quant aux résultats à obtenir en termes de processus et de livrables. Cette démarche pourrait convenir si le S.P.F. disposait déjà d'une expérience méthodologique pouvant être réutilisée dans les nouveaux processus. Cependant, cette expérience manque au S.P.F. et le risque existe que les méthodes de travail actuelles soient à nouveau introduites dans le cadre méthodologique proposé.

D propose une approche plus flexible, encadrée par RUP, basée sur la réalisation de projets pilotes et l'implémentation progressive des logiciels proposés. Ceci permettra d'impliquer au fur et à mesure les différents acteurs S.P.F. du projet, au moment où ils pourront concrètement participer aux projets. Cette approche progressive permettra également aux différents acteurs business et Ict d'assimiler graduellement les particularités des outils et méthodes proposées. On regrettera que la description des services proposés soit parfois insuffisamment détaillée.



#### **Critère 4 : Intégration Méthode – Outils (5%)**

Dans son offre, A présente quelques « modes d'emploi » des outils dans un petit nombre de cas d'utilisation. Ceux-ci ne permettent pas de juger dans quelle mesure l'intégration présente « out of the box » entre RUP et les outils de Rational est améliorée.

Dans l'offre B, l'intégration entre RUP et les outils Rational est clairement démontrée. Cependant, la maîtrise du mapping entre méthodologie et outils demande un apprentissage non négligeable.

L'offre C étant très ouverte au niveau méthodologique, l'intégration entre méthodologie et outils demandera une grande attention lors de la définition des livrables, afin de s'assurer d'une part qu'ils peuvent être produits par les outils proposés, et d'autre part que la paramétrisation et les développements sur mesure des outils correspondent bien aux livrables définis par la méthodologie. En ce sens, elle présente un risque plus important de divergence entre outils et méthode.

Dans l'offre D, la cohérence entre les livrables définis dans RUP et les logiciels proposés est présente. Il faudra cependant s'assurer que les exigences méthodologiques soient reflétées dans les outils, afin d'encadrer l'utilisation de ces derniers.

#### **Critère 5 : Lisibilité et conformité de l'offre (5 %)**

L'offre d'A est sommaire, passant d'un chapitre à l'autre d'une description de haut niveau de la solution à une description très détaillée du déroulement de certaines activités. Elle ne fait pas apparaître clairement l'engagement d'A par rapport à la mission proposée.

L'offre d'B est complète et décrit clairement les solutions proposées.

L'offre de C est d'un bon niveau. On peut regretter l'absence de détails sur certains points, dont les aspects d'intégration entre les composantes C et Open-Source de l'offre.

L'offre d'D est complète et décrit clairement les solutions proposées.

#### **Critère 6 : Case Study (5 %)**

Le case study, décrit dans l'annexe ..., avait pour objectif de permettre aux soumissionnaires de démontrer les caractéristiques de la méthodologie et des outils proposés dans le contexte du développement d'une application dans un environnement technique représentatif de celui du S.P.F. Les démonstrations se sont déroulées dans les locaux des soumissionnaires. Ceux-ci ont eu une semaine pour préparer leur présentation.

Le traitement du case study par A est largement insuffisant. A n'a pas été capable de présenter de manière cohérente les interactions entre méthodologie et outils, ni de présenter de manière convaincante les fonctionnalités élémentaire des outils proposés dans le contexte de l'exercice proposé. Aucun point du case study n'a été traité correctement. Le seul point positif de la session fut le retour d'expérience fourni par A (France) sur le projet Copernic au Ministère des Finances français. Ceci a amené la commission à attribuer une note inférieure à la cote d'exclusion définie au point 1.2.13.3.1 du cahier des charges.

Afin de traiter le case study, B a mis en place un environnement technique complexe représentatif de celui dans lequel ils seraient mis en œuvre au S.P.F. Ceci a permis à B de prouver sa capacité à implémenter les outils proposés dans cet environnement. B a abordé le case study de manière didactique et convaincante en illustrant les interactions entre méthodologie et outils. Cependant la présentation aurait pu prévoir une place plus importante pour la présentation des outils, au détriment de la présentation détaillée de la méthodologie. L'équipe réalisant le case study a fait preuve de sa grande connaissance des outils et méthodes en implémentant la plus grande partie de l'exercice proposé dans le temps imposé. On peut cependant regretter une mauvaise compréhension du change request 2 qui n'a pas permis de démontrer les fonctionnalités d'analyse d'impact existant dans l'outil.



Le traitement du case study par C fut inégal et marqué par un déséquilibre entre la présentation « statique » PowerPoint des possibilités de la solution et l'utilisation des outils dans le traitement de l'exercice proposé. Celui a été réalisé de façon très fragmentaire et n'a démontré qu'un sous-ensemble très limité des fonctionnalités des logiciels proposés. Les aspects relatifs au déploiement de la solution dans un environnement similaire à celui du S.P.F. n'ont pas été présentés. Les « change request » compris dans l'exercice n'ont été traités que sommairement.

Le traitement du case study par D fut très convaincant. L'infrastructure technique mise en place était représentative de l'architecture du S.P.F. Le case study a été utilisé comme fil rouge pour illustrer clairement les interactions entre méthode et outils. La présentation des outils, réalisée par une équipe maîtrisant outils et méthode, a permis de mettre en évidence leur facilité d'utilisation et les fonctionnalités pertinentes à la réalisation du case study. On regrettera le traitement sommaire du « change request » 2 relatif à l'analyse d'impact.

### **Critère 7 : Formation — Documentation (3 %)**

Les formations fournies par A sont insuffisantes et ne permettront pas aux utilisateurs de maîtriser la complexité des outils et de la méthodologie dans les délais proposés. Il est également à noter que l'offre de formation A regroupe en une seule formation les aspects outils et méthode de la solution proposée.

Les formations proposées par B sont complètes, le plan de formation défini est cohérent, leur contenu est détaillé. Le plan d'approche de la formation des formateurs est bien pensé et permettra d'atteindre les objectifs fixés.

Les formations proposées par C sont complètes. Cependant la structure des formations techniques est parfois étrange, notamment dans l'approche de la gestion de version qui comprise dans deux cours. L'approche de formation des formateurs est intéressante par son interactivité (système de « devoirs »)

Les formations proposées par D sont complètes et détaillées. Les formations « train the train » correspondent aux formations de certification proposées par les fournisseurs des logiciels proposés.

### **Critère 8 : SLA – Support (2 %)**

A donne une description très succincte du support fourni, les modalités d'accès ne sont pas détaillées : combien de personnes peuvent accéder au support ? ,... A ne fournit un support limité aux heures de bureau, même pour les incidents « bloquants.

B garantit un support de bonne qualité, y compris 7J/7J 24h/24h pour les incidents de type bloquant. On regrettera qu' B renvoie à son site Web pour le détail des modalités de support.

Les modalités de support proposées par C sont peu satisfaisantes : la mise à disposition d'un niveau de support acceptable nécessite la mise en place d'un Technical Account Manager dont le prix n'est pas compris dans les licences et dont les modalités d'utilisation sont restrictives. De plus, C impose un préavis de 15 jours et des minima de durées de mission de 5 jours pour l'accès au coach.

Le support aux outils fournis par D varie de bon, pour les produits Borland et Mercury à faible pour les produits Embarcadero, où les fenêtres de support sont limitées. Les modalités de support sont détaillées dans l'offre. D s'engage à jouer son rôle d'intégrateur et de support de première ligne.



## 12.2. Les outils de développement et de modélisation, soutiens à la méthodologie

Cette annexe est une compilation des informations glanées dans l'offre remise par le soumissionnaire lauréat.

### La solution logicielle retenue

Ces logiciels permettent l'analyse et le développement d'applications informatiques, depuis la phase de capture des spécifications jusqu'aux tests des applications.

Cette suite logicielle offre les fonctionnalités suivantes en concordance avec le cahier des charges :

- ↪ Capture des spécifications.
- ↪ Analyse et Design UML.
- ↪ Programmation Java - J2EE.
- ↪ Modélisation de données.
- ↪ Support au test d'applications et contrôle de qualité.
- ↪ Dictionnaire de données – Support collaboratif.

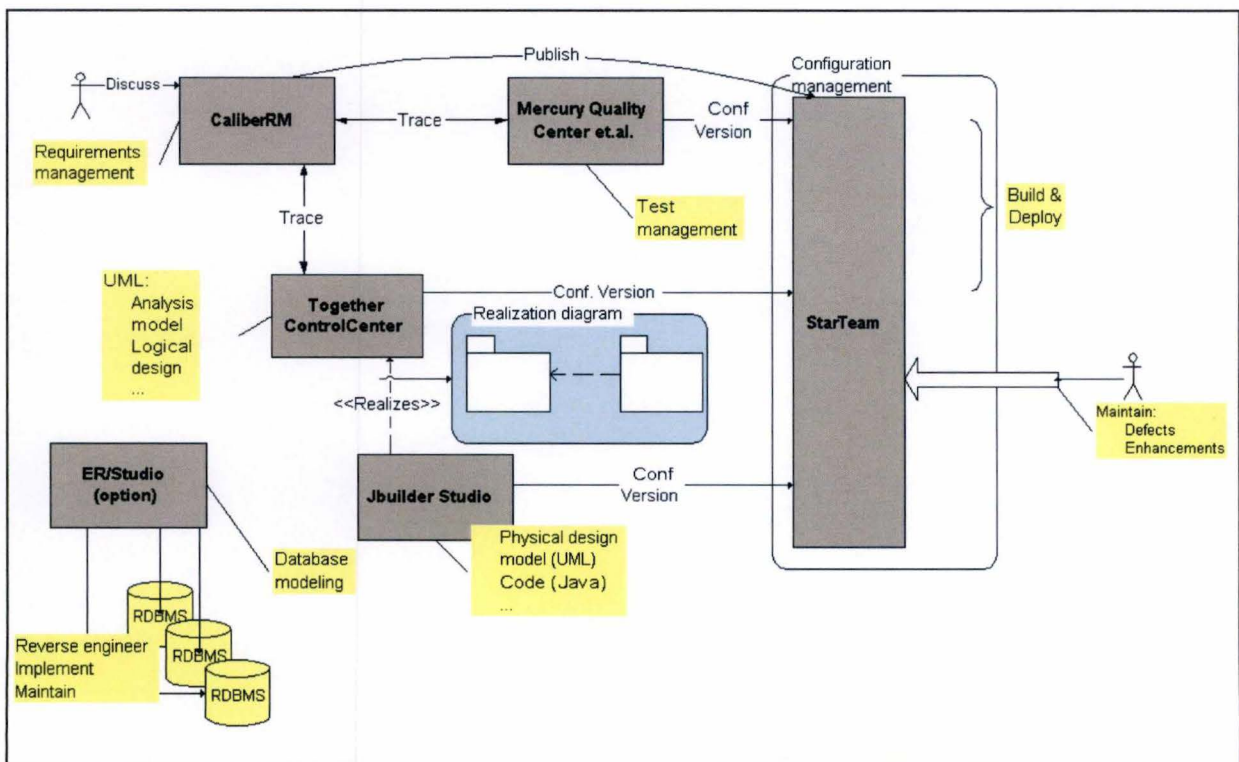


Figure 12-1 : La suite logicielle retenue

### Capture et gestion des spécifications : Borland CaliberRM

#### Objectifs :

- ↳ Permettre aux utilisateurs Business et ICT de tracer l'utilisation de ces spécifications au cours du développement du système d'information.
- ↳ Disposer de facilités de reporting accessibles aux utilisateurs Business et ICT
- ↳ Etre compatible avec l'environnement bureautique standard du S.P.F.

### **Avantages :**

CaliberRM est une solution de gestion des besoins multi-plate-forme à la base des développements de logiciels. Cette solution s'est imposée grâce à capacité de mise en œuvre transparente, une grande simplicité d'emploi et de faibles coûts de formation, avec à la clé un coût d'acquisition total en baisse.

Point de départ essentiel pour l'intégration d'activités participant d'une bonne gestion du cycle de vie des applications, CaliberRM est intégré dans la gamme de produits ALM de Borland, de sorte que les exigences sont visibles et traçables tout au long du processus de développement des applications.

### **CaliberRM 6.0 :**

- ↳ Dispose d'une prise en charge améliorée de la personnalisation et des intégrations en temps réel au travers d'une nouvelle architecture événementielle
- ↳ Inclut les fonctions d'authentification LDAP (Lightweight Directory Access Protocol) afin de gérer les informations de connexion de toutes les applications et de façon centralisée
- ↳ Assure une intégration étroite avec Borland StarTeam 6.0, qui améliore la prévisibilité, le contrôle et la qualité du processus de développement logiciel.
- ↳ Est à présent intégré à l'environnement de développement Borland et dans une mise à jour de TestDirector 8.0, la solution de test de Mercury Interactive

## **Analyse/Design UML: Borland Together Control Center**

### **Objectifs :**

- ↳ Permettre l'exploitation des spécifications définies au moyen de l'outil de capture de spécifications décrit plus haut, ainsi que la traçabilité par rapport à celles-ci.
- ↳ Permettre la réalisation, au moyen d'une interface graphique, des diagrammes définis par UML 1.4
- ↳ Faciliter la génération de code J2EE à partir des schémas UML
- ↳ Offrir des fonctionnalités de reporting étendues.

### **Avantages :**

Borland Together accélère le cycle de développement des applications par une intégration étroite de l'environnement de développement avec une solution centrée sur la conception construite pour modéliser visuellement, mesurer la qualité, et améliorer la productivité de l'équipe.



Grâce à ses outils de modélisation et de visualisation du code source, Together réduit significativement la complexité technique grâce à des modèles éprouvés de conception qui minimisent les risques d'erreur tout au long du cycle de vie de l'application.

Together permet aux équipes de développement de synchroniser le modèle et le code source du projet, de générer la documentation de l'application, de prendre en compte la qualité et le code source du projet et de s'interfacer avec tous les EDI majeurs du marché Jbuilder, Visual Studio NET, Websphere, etc.).

### Borland Together Architecte

Borland Together Architecte est une solution de modélisation globale, de conception visuelle UML (modeling language) destinée à ceux qui développent des architectures applicatives d'entreprise et doivent échanger avec les intervenants de diverses entités techniques ou fonctionnelles.

### Borland Together Designer

Borland Together Designer est une solution de modélisation UML inter plate-forme destinée aux analystes fonctionnels et aux contextes où la formalisation graphique peut optimiser la définition des contraintes et clarifier les échanges sur l'architecture et le code applicatif.

La capacité de créer des modèles uniquement à partir d'un langage de modélisation, de les transformer aisément en des modèles spécifiques à la plate-forme doit faciliter l'agilité des applications afin qu'elles répondent aux changements constants.

L'édition Designer bénéficie du support d'OCL (Object Constraint Language) et UML 2.0 pour spécifier les exigences métier et techniques au moyen de modèles UML.

### Borland Together Developer

Borland Together Développeur propose un environnement exclusif de modélisation centré sur le code permettant aux développeurs de réduire la complexité des applications grâce à l'utilisation des classes UML et des diagrammes de séquence.

Together Developer dispose d'intégrations aux EDI afin que les développeurs puissent comprendre plus simplement les éléments conceptuels de l'application en transformant/visualisant le code.

La technologie Together LiveSource synchronise automatiquement les modèles et le code de programmation pour garantir qu'ils sont toujours en phase

## **Programmation J2EE: Borland Enterprise Studio**

### **Objectifs :**

- ↳ Ecrire le code J2EE des applications
- ↳ Permettre la traçabilité par rapport aux diagrammes et objets créés dans les phases précédentes.
- ↳ Exploiter les diagrammes UML réalisés au cours des phases d'analyse et de design
- ↳ Offrir un environnement de développement intégré, avec une interface utilisateur performante.

- ↪ Offrir des fonctions avancées de support au développeur (bibliothèque de design patterns,...)
- ↪ Permettre l'interaction bidirectionnelle entre diagrammes UML et code Java
- ↪ Supporter les API J2EE 1.4 dans un environnement WebLogic
- ↪ Fournir des fonctionnalités de build d'applications complexes
- ↪ Permettre le reverse-engineering de projets J2EE développés dans un environnement Jbuilder
- ↪ Permettre le déploiement d'applications dans les environnements de test, intégration, acceptation, et production Weblogic

### **Avantages :**

Borland Enterprise Studio (B.E.S.) fournit des fonctionnalités avancées de modélisation, qui aident les équipes à mieux comprendre et à mieux communiquer les besoins et la structure des applications.

De plus les audits et les métriques aident les équipes à vérifier la conformité aux standards tout en respectant les meilleures pratiques du développement logiciel.

Débogage complet, test unitaire, garantie des performances, surveillance des fuites mémoire, tout contribue à améliorer la qualité, à raccourcir les cycles du contrôle qualité et à remplir les objectifs de performance.

La version évoluée de Jbuilder, environnement multi plate-forme de développement intégré pour Java, comporte une centaine de fonctionnalités qui visent à améliorer la productivité des équipes en entreprise et des développeurs Java.

Comme par exemple, une nouvelle interface utilisateur avec plusieurs fenêtres "ancrables", des "personnalités" configurables par le développeur, un code présenté en mode "portefeuille", la prise en charge du "refactoring" ; ou encore l'intégration d'un concepteur Struts pour le développement d'applications Web évoluées, un concepteur visuel de services web, et d'autres fonctionnalités pour le développement d'applications J2EE.

Grâce à ses modules de conception bidirectionnels et de déploiement accéléré sur les principaux serveurs d'applications J2EE™ (comme BEA WebLogic), JBuilder 2005 Entreprise aidera à concevoir plus rapidement des EJB (Enterprise JavaBeans), des solutions et services Web et des applications XML, mobiles ou pour bases de données.

### **Principales fonctionnalités :**

- ↪ Editeur supportant Unicode avec gestionnaire de projet, compilateur, débogueur
- ↪ Composants JavaBeans dbSwing orientés données pour le développement base de données
- ↪ Editeur et concepteur graphique de flux JavaServer Faces
- ↪ Audits de code Outils additionnels pour la gestion des performances
- ↪ Refactoring distribué
- ↪ Support avancé de gestionnaires de versions et intégration de Subversion



- ↪ Outils avancés de développement Web
- ↪ Profilage J2EE avec Optimizeit Request Analyzer
- ↪ Concepteur EJB J2EE 1.4
- ↪ Déploiement incrémental sur serveurs d'applications J2EE
- ↪ Liaison directe avec les exigences depuis CaliberRM
- ↪ Intégration étendue de la gestion du cycle de vie applicatif

### **Modélisation DB : Embarcadero E/R Studio**

#### **Objectifs :**

- ↪ permettre la modélisation de DB relationnelles
- ↪ permettre la constitution d'un dictionnaire de données
- ↪ Permettre la modélisation relationnelle au niveau conceptuel
- ↪ Permettre la modélisation conceptuelle de données à partir de modèles UML
- ↪ Permettre la gestion des conversions entre les diagrammes de classe UML et les diagrammes Entité - Relations.
- ↪ Permettre le reverse engineering automatique de structures de données relationnelles existantes
- ↪ Permettre le reverse engineering, manuel ou automatique, de structures de données non-relationnelles (IDS II, DL/1, ISAM), au moins sous forme documentaire.

#### **Avantages :**

E/R Studio est une application de modélisation de données destinée à la conception et la réalisation de bases de données logiques et physiques.

Son environnement de conception multi-niveaux répond aux besoins quotidiens des administrateurs de bases de données, des développeurs et des architectes de données chargés de la conception et de la maintenance d'applications de base de données volumineuses et complexes.

L'interface et les processus progressifs d'ER/Studio apportent une solution efficace aux problèmes de simplicité d'utilisation dont souffrent toujours les outils de modélisation de données.

L'application permet aux utilisateurs de créer, de comprendre et de gérer la modélisation de bases de données vitales au sein d'une entreprise.

Elle supporte le cycle de vie complet des applications grâce à un certain nombre de fonctions, parmi lesquelles:

- ↪ de puissantes capacités de modélisation logique et physique
- ↪ synchronisation bidirectionnelle des modèles logique et physique, et entre modèles et base de données
- ↪ une architecture ouverte permettant d'étendre les fonctionnalités du produit

- ↳ des fonctions de génération de rapports et de documentation en langage HTML, vous permettant de communiquer les différents travaux au sein de l'entreprise

Exemples de fonctions :

- ↳ Reverse Engineering - Génération de code Java pour prototyper les applications
- ↳ Mise à jour modèle/base de données
- ↳ Documentation et Reporting automatisés (HTML, Word, ASCII)

### **Tests d'applications : Mercury Suite**

#### **Objectifs :**

- ↳ Permettre la définition et la gestion de scripts de tests.
- ↳ Permettre la traçabilité des tests depuis les spécifications de l'application jusqu'au code source de celle-ci.
- ↳ Permettre le support des tests fonctionnels, de performance, d'intégration, de régression, ainsi que le test de composants.
- ↳ Supporter les tests en environnement local et distant.
- ↳ Fournir des rapports étendus quant aux résultats des tests effectués Permettre le support des tests fonctionnels, de performance, d'intégration, de régression, ainsi que le test de composants

#### **Avantages :**

Mercury Quality Center est une solution WEB intégrée et complète, qui offre les fonctions d'assurance qualité sur un large éventail d'environnements informatiques et d'applications.

Elle comprend une suite intégrée d'applications et de meilleures pratiques basées sur les fonctions ainsi qu'une base ouverte, évolutive et étendue.

Tous ces outils visent à optimiser et automatiser les principales activités de gestion de la qualité, notamment la gestion des besoins, des tests, et des anomalies, aussi bien pour les tests fonctionnels que les tests des processus métier.

#### **Mercury TestDirector**

Mercury TestDirector permet de déployer rapidement et efficacement des applications de qualité en offrant des procédures cohérentes et reproductibles qui permettent de recueillir les besoins, de planifier et de programmer les tests, d'analyser les résultats et de gérer les anomalies et les problèmes.

Avec TestDirector, différentes entités de l'entreprise peuvent contribuer au processus de qualité :

- ↳ Les analystes métier définissent les besoins des applications et les objectifs des tests
- ↳ Les responsables des tests et les chefs de projet conçoivent des plans de test et développent des cas de test



- ↳ Les ingénieurs en automatisation des tests créent des scripts automatisés et les stockent dans le référentiel
- ↳ Les testeurs d'assurance qualité exécutent les tests manuels et automatisés, indiquent les résultats d'exécution et saisissent les anomalies
- ↳ Les développeurs révisent et réparent les anomalies enregistrées dans la base de données
- ↳ Les chefs de projet créent des rapports sur l'état de l'application et gèrent l'affectation des ressources
- ↳ Les responsables du produit décident quand une application est prête à être déployée.

### Mercury Application Delivery Dashboard

L'application Delivery Dashboard offre des indicateurs de performance clés pour tous les projets liés à la qualité et aux performances, fournissant ainsi aux dirigeants et aux professionnels des services informatiques, préalablement au déploiement, une visibilité en temps réel quant à la santé des applications et de l'infrastructure.

Ce tableau de bord vous fournit la visibilité et les informations requises pour prendre une décision rationnelle au moment du lancement d'une application.

Enfin, en matière d'anomalies fonctionnelles, Application Delivery Dashboard ne se contente pas de notifier les anomalies restantes dans chaque niveau de gravité ; il indique en temps réel si vous trouvez plus d'anomalies que vous n'en résolvez.

### Mercury Functional Testing

Mercury Functional Testing permet de profiter des ressources de test offertes par les scripts. Ainsi les ingénieurs qualité peuvent utiliser Mercury Functional Testing pour créer des « scripts composés » qui comportent des tests élaborés.

Mercury Functional Testing tire profit de l'intégration de WinRunner et de QuickTest Professional : chaque produit peut appeler les scripts de l'autre produit, et les résultats des tests apparaissent dans une interface commune

Mercury Functional Testing répond aux besoins des experts techniques, des experts du domaine ou des analystes métier, ce qui permet de déployer des applications de meilleure qualité plus rapidement, avec un coût et des risques moindres.

Mercury Functional Testing présente ainsi de nombreux avantages pour le service d'assurance qualité :

- ↳ Il permet à toute l'équipe de créer des suites de tests sophistiquées avec une formation minimale.
- ↳ Il fonctionne correctement avec tous les environnements, les fichiers et les processus métier.
- ↳ Il documente et reproduit les anomalies pour les développeurs, afin de répondre aux délais de production.
- ↳ Il simplifie les tests de non-régression pour les applications et les environnements en constante évolution.

### Mercury QuickTest Professional

Le logiciel QuickTest Professional offre de nombreux avantages pour le service d'assurance qualité :

- ↳ Il permet à toute l'équipe de créer des suites de tests sophistiquées avec une formation minimale.
- ↳ Il fonctionne avec tous les environnements, les fichiers et les processus métier.
- ↳ Il documente et reproduit les anomalies pour les développeurs, leur permettant d'éliminer plus rapidement les anomalies et donc de respecter les délais de production.
- ↳ Il simplifie les tests de non-régression pour les applications et les environnements en constante évolution.

### Mercury WinRunner

Ce logiciel réduit le temps consacré aux tests grâce à l'automatisation des tâches répétitives et à l'optimisation des efforts de tests.

C'est un outil puissant qui permet d'effectuer des tests fonctionnels et de non-régression à l'échelle de l'entreprise.

WinRunner localise, vérifie et relit automatiquement les interactions des utilisateurs, pour que vous puissiez identifier les anomalies et vous assurer que les processus métier fonctionnent parfaitement tout au long du déploiement et restent fiables.

### Mercury Performance Center, principalement LoadRunner :

L'outil de test LoadRunner permet de connaître le comportement du logiciel, et ses performances, en simulant l'entreprise, c'est à dire les nombreux utilisateurs différents.

Cet outil sert dans plusieurs domaines car il reproduit le comportement d'un utilisateur afin d'observer les réactions du système. Il crée des utilisateurs virtuels grâce à LoadRunner Virtuel User Generator, puis enregistre les procédés habituels de transactions et programme les tests associés. Ces tests pourront être utilisés grâce à des jeux de variables qui pourront être stockées dans un fichier (EXCEL par exemple) ou dans une base de données.

Le test est ainsi plus réaliste et significatif.

De plus, la fréquence des transactions, le nombre d'entrées et même le temps de réflexion de l'utilisateur peuvent être modifiés.

LoadRunner prévoit aussi la simulation de plusieurs clients ayant des adresses IP différentes. Une fois les utilisateurs créés, LoadRunner Scenario Wizard permet de composer divers scénarios, et de les modifier, pour par exemple référencer les performances d'autres sites ou d'autres bases de données.

La supervision du test, (organisation du scénario...), comme l'observation des résultats, peut se faire grâce à un écran de contrôle en temps réel (Visual Controller) qui permet entre autre de localiser les goulots d'étranglement.



Les questions habituelles lorsqu'une application DB2 est développée concernent non seulement la rapidité d'exécution lorsque les clients connectés sont très nombreux, mais aussi l'organisation de la base de données.

De plus, les applications DB2, qui s'adaptent à plusieurs environnements, ont souvent besoin d'être transportées d'un environnement à un autre, et cette manipulation peut en dégrader les performances. LoadRunner permet de faire face à ce problème, et de développer l'application en toute confiance.

### **Support collaboratif : Borland Star Team**

#### **Objectifs :**

- ↳ Permettre la consolidation, le partage, le suivi et la diffusion des deliverables réalisés
- ↳ Permettre la gestion des accès multiutilisateurs aux modèles; la gestion de groupes d'utilisateurs,...
- ↳ Assurer la gestion des versions des différents modèles et sources créés par les logiciels
- ↳ Permettre le suivi des anomalies détectées dans les applications et de leur correction.
- ↳ Permettre la consolidation des différents modèles dans un repository central.
- ↳ Permettre la réalisation d'analyse d'impact lors de la modification de modèles.
- ↳ Permettre la réalisation et la gestion d'un dictionnaire de données centralisé.
- ↳ Permettre la reprise de modèles existants (notamment ceux produits par les outils XDE de Rational,...)

#### **Avantages :**

Cette solution de gestion des configurations logicielles et de conduite du changement procure un niveau accru de sécurité et de collaboration ainsi que de meilleurs délais de réalisation.

Alors que les entreprises évaluent la nécessité d'externaliser leurs activités de développement, StarTeam 6.0 améliore la gestion des activités de développement logiciel entre équipes et sites géographiquement dispersés. Il résout en effet les difficultés de communications inhérentes aux projets de développement qui risqueraient de réduire leur productivité. StarTeam 6.0 constitue la partie « gestion » du portefeuille multi plate-forme de solutions de gestion du cycle de vie des applications proposé par la suite des outils Borland.

#### **Une sécurité accrue :**

StarTeam 6.0 prend en charge les tâches d'authentification via Microsoft Active Directory, une mise en œuvre du protocole LDAP (Lightweight Directory Access Protocol).

Cette solution permet à un administrateur système de gérer l'authentification des utilisateurs pour plusieurs configurations de serveurs à partir d'une source unique.

Les équipes distribuées connaissent souvent des interruptions temporaires de connexion réseau qui ont un impact sur leur productivité. Les nouvelles fonctionnalités de StarTeam 6.0 assurent une

reconnexion automatique en cas d'interruption de la connexion réseau, ce qui évite de relancer les applications.

De plus, les fonctions de diagnostic évoluées assurent un suivi ciblé des problèmes permettant de réduire le temps nécessaire pour les résoudre. Les nouvelles options de fichiers de log proposées par StarTeam 6.0 sont conçues pour aider les utilisateurs à spécifier les critères de suivi des problèmes et définir les mesures à prendre.

Le client multi plate-forme de StarTeam 6.0 permet également aux utilisateurs de choisir une interface qui leur convient. En effet, les différences subtiles entre les types de commandes disponibles dans les nombreux systèmes d'exploitation peuvent gêner les utilisateurs. En modifiant l'aspect des commandes au sein du client multi-plate-forme, ce dernier permet aux utilisateurs de travailler dans un environnement familier.





### 12.3. Les conventions de nommage



Données - Convention de nommage

Données

Convention de nommage

#### LISTE DE DISTRIBUTION

Compagnie	Nom	Responsabilité	Pour Info / Pour Acceptation
SPF Finances	Tous les membres de STIR		Pour info
	Tous les membres de SupDev		Pour info
STEER	Tous les membres de STEER		Pour info





Données - Convention de nommage

## VISAS et ACCEPTATIONS

	Nom	Date	Signature
Préparé par :	Vincent PECHON – SPF Finances		
Avec la collaboration active de	Jean-Christophe TROUSSART Patrick Syemons		
Accepté par :	Didier Leemans – SPF Finances		
Accepté par :	Nicole Vereruyse – SPF Finances		
Accepté par :	Hilde Aerts – SPF Finances		

## HISTORIQUE DU DOCUMENT

Version	Date	Auteur	Description
01	29/03/2005	PECHON	Création
02	30/03/2005	PECHON	Mise en conformité SupDev avec la collaboration de Jean-Christophe TROUSSART
03	22/04/2005	PECHON	Ajout type de données « Devises Etrangères »

26/05/2006  
Convention\_de\_nommage-2[1].52

Page 2 de 15



## TABLE DES MATIERES

<b>1. INTRODUCTION.....</b>	<b>4</b>
<b>2. LE SCHÉMA CONCEPTUEL.....</b>	<b>5</b>
2.1. Les classes.....	5
2.1.1. Aperçu général .....	5
2.1.2. Exemples .....	5
2.1.3. Le dictionnaire des données .....	5
2.1.4. Le dictionnaire des acronymes.....	6
2.1.5. Exemples .....	6
2.2. Les attributs de classe .....	7
2.2.1. Aperçu général .....	7
2.2.2. Champs x : .....	7
2.2.3. Champs Cb..b .....	8
2.2.4. Champs Mn..n: .....	8
2.2.5. Sémantique du nom de l'attribut .....	8
2.2.6. Exemple:.....	8
2.2.7. Liaison classe-attributs .....	9
<b>3. LES BASES DE DONNÉES .....</b>	<b>11</b>
3.1. Les tables.....	11
3.1.1. Aperçu général .....	11
3.1.2. Exemples .....	11
3.2. Les colonnes .....	12
3.2.1. Aperçu général .....	12
3.2.2. Champs X : .....	12
3.2.3. Champs CB..B et MN..N: .....	13
3.2.4. Exemple:.....	13
3.2.5. Liaison table-colonne .....	14





## 1. Introduction

Ce document a pour but de définir au sein du SPF Finances une convention de nommage pour les données définies dans les nouvelles applications.

Pour des raisons techniques et pratiques, il ne nous est pas permis d'utiliser la dénomination bilingue complète des données.

De ce fait, nous aurons recours à une convention se basant sur une dénomination abrégée des données. Une attention toute particulière sera portée à ces abréviations, de telle manière qu'elles soient le plus explicite possible.

La convention qui est décrite ci-dessous est liée à l'obligation de tenir un dictionnaire des abréviations (appelé dictionnaire des acronymes) et un dictionnaire des données lié au glossaire.

Ces dictionnaires comprendront deux parties distinctes :

1. traduction bilingue :  
Cette partie donnera la traduction complète en français et en néerlandais de l'entrée du dictionnaire.
2. définition bilingue  
Cette partie donnera la traduction complète en français et en néerlandais de la définition de l'entrée du dictionnaire.

Le but de ces dictionnaires est de maintenir la cohérence des données et de leur nomination. Une saine gestion en sera assurée par un « gestionnaire de dictionnaire » qui en contrôlera les entrées et les validera.

Le rôle de cette personne est primordial, tant d'un point de vue business qu'IT. Il devra être désigné avant toute utilisation des outils de modélisation de données pour les nouvelles applications

Les présentes conventions ont été établies en tenant compte des critères suivants :

- Souplesse d'utilisation ;
- Facilitation du suivi et de la maintenance des modélisations ;
- Lisibilité aisée après un temps d'adaptation et d'écologie ;
- Traçabilité complète des noms depuis la modélisation jusqu'à la programmation et les bases de données.



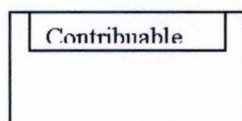
Attention : les acronymes et noms utilisés dans ce document le sont à titre d'exemple et reflètent pas nécessairement le contenu du dictionnaire des données en production.

## 2. Le schéma conceptuel

### 2.1. Les classes

Une classe est une représentation abstraite d'un concept particulier tel qu'un contribuable, un impôt, une dette, .....

En UML une classe est représentée comme suit :



#### 2.1.1. Aperçu général

Les noms des classes auront le format suivant :

**KMnn...nMnn...n**

Avec

- K = première lettre majuscule constante et obligatoire montrant qu'il s'agit d'une classe
- M = représente une lettre majuscule quelconque
- n = représente une lettre minuscule quelconque
- Mnn...nMnn...n: nom de la classe au singulier et de longueur libre. Dans le cas où plusieurs mots composent le nom de la classe, ces différents noms seront accolés sans espaces et commenceront tous par une majuscule.

#### 2.1.2. Exemples

La classe « personne » sera nommée **KPersonne**

La classe « situation juridique » sera nommée **KSituationJuridique**

#### 2.1.3. Le dictionnaire des données

Chaque classe sera reprise dans le dictionnaire des données, sa dénomination sera traduite en français et en néerlandais et recevra une définition bilingue.





#### *2.1.4. Le dictionnaire des acronymes*

Chaque mot composant le nom d'une classe recevra une abréviation (acronyme) de 3 à 5 positions. Ainsi, l'acronyme général d'une classe se présentera comme suit :

**Cbb....b**

- C = représente une lettre majuscule quelconque
- b = représente une lettre minuscule quelconque ou un chiffre
- Cb...b : abréviation de la classe. Seule la première lettre de cette abréviation est une majuscule. Le nom de la classe peut être composé de plusieurs acronymes.

Dans le dictionnaire des acronymes, on traduira de manière **bilingue** la signification de l'acronyme.

#### *2.1.5. Exemples*

La classe « personne » serait nommée **KPersonne**, son abréviation pourrait être « **Pers** »

La classe « situation juridique » serait nommée **KSituationJuridique**, son abréviation pourrait être « **Sitjur** ». Il est à noter que dans ce cas, l'abréviation « Sitjur » devra également se retrouver dans le dictionnaire des acronymes.



## 2.2. Les attributs de classe

Un attribut d'une classe est une propriété de cette classe. Ainsi, dans la classe KContribuable, ce qui caractérise un contribuable, c'est son nom, son prénom, son adresse, sa profession, ...

KContribuable
Nom
Prénom

### 2.2.1. Aperçu général

Les noms des attributs auront le format suivant :

XCb...bMn..nMn..n...Mn..n

Avec

- x = première lettre minuscule obligatoire désignant le type de donnée.
- M = représente une lettre majuscule quelconque
- n = représente une lettre minuscule quelconque ou un chiffre
- Cb..b = abréviation de la classe d'origine de la donnée qui peut être composée de plusieurs acronymes
- Mn..nMn..n = nom de la donnée, éventuellement composé de plusieurs abréviations pouvant aller de 3 à 5 positions

### 2.2.2. Champs x :

Ce champ comprendra une et une seule lettre qui définira le type général de la donnée.

Les différents types seront les suivants :

- t – Texte
- d – Date
- c – Code
- m – Montant en €
- f – Montant en devises non €
- n – Nombre
- b – Contenu non structuré
- p – Pourcentage
- o – Ordre (séquence)





- s – Time Stamp (date, heure, minute, seconde)
- l – Logique (oui/non, vrai/faux)

### 2.2.3. Champs Cb..b

Ce champ est prévu pour indiquer l'acronyme de la classe dans laquelle la donnée a été initiée. (voir liaison classe attribut)

### 2.2.4. Champs Mn..n:

Ces champs sont prévus pour les abréviations de dénomination de la donnée; la première lettre de chaque abréviation est une majuscule. Les abréviations pourront avoir une longueur de 3 à 5 positions.

Le nom d'un attribut peut être une combinaison d'abréviations, afin de traduire au mieux la sémantique à accorder au nom. Chacune des abréviations employées doit obligatoirement être définie dans le dictionnaire des acronymes.

### 2.2.5. Sémantique du nom de l'attribut

La structure répond à la philosophie OO. Ainsi les données sont nommées selon le principe « du général au particulier ».

Le nombre d'abréviations liées au nom d'une donnée peut être variable en fonction de la sémantique voulue à la dénomination.

### 2.2.6. Exemple:

Le nom de la rue où est domicilié un mandataire serait représenté comme suit :

tMdtAdrRueNom

Avec « t » pour indiquer qu'il s'agit d'un champ texte ;  
« Mdt » pour mandataire ;  
« Adr » pour adresse ;  
« Rue » pour rue ;  
« Nom » pour nom de la rue.

Il faut donc lire ce code de droite à gauche pour en définir la sémantique => le nom de la rue de l'adresse du mandataire

Les données complètes de l'adresse d'un mandataire seraient les suivantes :



tMdtAdrRueNom    => le nom de la rue de l'adresse du mandataire  
nMdtAdrRueNum    => le numéro dans la rue de l'adresse du mandataire  
cMdtAdrCps        => le code postal de l'adresse du mandataire  
tMdtAdrLoc        => le nom de la localité de l'adresse du mandataire  
tMdtAdrPay        => le nom du pays de l'adresse du mandataire

Le nom du mandataire serait : tMdtNom

Cas particulier : les noms d'attributs de type f (devises non €) devront se terminer par une indication du nom de la devise attendue.

### 2.2.7. Liaison classe-attributs

Comme il a déjà été dit ; le nombre d'abréviations liées au nom d'une donnée peut être variable en fonction de la sémantique voulue mais varie aussi en fonction de la classe à laquelle la donnée est attachée.

Comme précisé ci avant, chaque classe représente un concept unique. Ceci implique que les données définissant ce concept et n'ayant aucune liaison avec une autre classe sont également uniques dans leur conception.

Ainsi, dans la classe KMandataire, le nom d'un mandataire est directement dépendant de cette classe.

Le formalisme de nommage sera alors le suivant :

**Le nom de la donnée doit comporter dans sa dénomination l'acronyme du nom de la classe générique à laquelle il se rapporte.**

Analysons à présent la **classe mandataire** (kMandataire, dont l'acronyme est Mdt) et les données relatives au nom et à l'adresse d'un mandataire reprises dans cette classe.

KMandataire
tMdtNom
tMdtAdrRueNom
nMdtAdrRueNum
cMdtAdrCps
tMdtAdrLoc
tMdtAdrPay

Analysons à présent la **classe situation juridique** (kSituationJuridique, dont l'acronyme est Sitjur)

Le code de la situation juridique serait : cSitjurCode

26/05/2006

Convention\_de\_nommage-2[1].52

Page 9 de 15





De cette manière, lors de l'emploi d'une de ces données dans une autre classe, sous forme d'attribut ou de lien interclasse, leur origine sera connue et leur définition dans le dictionnaire des données sera commune.

Par exemple, prenons la classe KClient, définissons le nom et l'adresse complète du client et mettons-y le nom de son mandataire.

KClient
tCliNom
tCliAdrRueNom
nCliAdrRueNum
cCliAdrCps
tCliAdrLoc
tCliAdrPay
<u>tMdtNom</u>

De cette manière, on voit directement que le nom du mandataire est lié à la classe mandataire et qu'il ne fait pas partie des données « propres » définissant la classe KClient.

Une autre représentation serait de lier les classes.

KMandataire
tMdtNom
tMdtAdrRueNom
nMdtAdrRueNum
cMdtAdrCps
tMdtAdrLoc
tMdtAdrPay

KClient
tCliNom
tCliAdrRueNom
nCliAdrRueNum
cCliAdrCps
tCliAdrLoc
tCliAdrPay

Cette représentation signifie qu'il existe une relation entre un Mandataire et un Client.

### 3. Les bases de données

Les données identifiées lors des phases d'analyses en amont seront persistées dans les bases de données du RDC. Des conventions de nommages basées sur les mêmes principes que ceux exposés ci-avant s'appliqueront aux tables et colonnes de ces bases de données RDC.

Ce sont ces conventions de nommage qui seront exposées ici, pour plus de détails techniques spécifiques à DB2, il est renvoyé au document *Guidelines Developpers* qui peut être trouvé à <http://kadscan2/fm/explorerxp/>.

#### 3.1. Les tables

##### 3.1.1. Aperçu général

Les noms des tables auront le format suivant : MNN...N\_MNN...N

Avec

- MNN...NMNN...N := représente le nom de la table au singulier et en majuscules. Dans le cas où plusieurs mots composent le nom de la table, ces différents noms seront séparés par le caractère '\_'.
- Les noms donnés aux tables sont libres, pour autant qu'ils respectent les prescrits techniques repris dans le document *Guidelines Developpers* du RDC (caractères autorisés, ne pas commencer par un chiffre, mots réservés, max 128 de long, ...).
- Eviter autant que possible les abréviations pour les noms des tables, utiliser "ADRESSE" plutôt que "ADR". Cela simplifie la lecture surtout dans les environnements multilingues.
- Si le nom de la table contient plusieurs mots (l'objet principal et le(s) qualifiant(s)), toujours mettre l'objet principal en tête et le(s) qualifiant(s) ensuite. Par exemple "DOSSIER\_OUVERT", "DOSSIER\_SURVEILLE", "DOSSIER\_HISTORIQUE". Cette règle s'appliquera indépendamment de la langue et de la correction linguistique. Ainsi on utilisera "BESTELLING\_OPEN" ou "COMMANDE\_OUVERTE", de même on utilisera "APPLICATIE\_TYPE" ou "APPLICATION\_TYPE".
- Les tables représentant des relations seront nommées en concaténant le nom des tables présentes dans la relation. Par exemple, avec les tables "DOCTEUR" et "PATIENT", si l'on accepte qu'un docteur ait plusieurs patients et qu'un patient peut avoir plusieurs docteurs, la table qui implémentera cette relation sera nommée "DOCTEUR\_PATIENT".

##### 3.1.2. Exemples

La table « personne » sera nommée **PERSONNE**.

La table « situation juridique » sera nommée **SITUATION\_JURIDIQUE**.





### 3.2. Les colonnes

#### 3.2.1. Aperçu général

Les conventions de nommage s'appliquant aux colonnes des tables relationnelles suivent la même logique que celle établie pour les conventions de nommage des attributs de classe (point 2.2)

Les noms des colonnes auront le format suivant :

X\_CB..B\_MN..N\_...\_MN..N

Avec

- X = première lettre majuscule obligatoire désignant le type de donnée.
- C, B, M et N représentent des lettres majuscules quelconques
- CB..B = abréviation de la classe d'origine de la donnée
- MN..N\_MN..N = nom de la donnée, éventuellement composé de plusieurs acronymes pouvant aller de 3 à 5 positions Ces acronymes seront séparés par le caractère '\_'

#### 3.2.2. Champs X :

Ce champ comprendra une et une seule lettre qui définira le type général de la donnée.

Les différents types seront les suivants :

- T – Texte
- D – Date
- C – Code
- M – Montant en €
- F – Montant en devises non €
- N – Nombre
- B – Contenu non structuré
- P – Pourcentage
- O – Ordre (séquence)
- S – Time Stamp (date, heure, minute, seconde)
- L – Logique (oui/non, vrai/faux)



### 3.2.3. Champs CB..B et MN..N:

Ces champs sont prévus pour les abréviations de dénomination de la donnée; tous les caractères seront des majuscules. Les abréviations pourront avoir une longueur de 3 à 5 positions.

Le nom d'une colonne peut être une combinaison d'abréviations, afin de traduire au mieux la sémantique à accorder au nom. Chacune des abréviations employées doit obligatoirement être définie dans le dictionnaire des acronymes. La première abréviation se rapporte à la classe d'origine de la colonne.

On utilisera pour les bases de données les mêmes abréviations que celles utilisées dans les attributs des classes.

Etant donné que la logique suivie est la même que celle des attributs des classes, la structure suivra également la philosophie OO qui consiste à nommer selon le principe « du général au particulier ». Le nombre d'abréviations liées au nom d'une donnée peut être variable en fonction de la sémantique voulue pour la dénomination.

### 3.2.4. Exemple:

On s'efforcera dans un souci de traçabilité de faire correspondre les noms des colonnes avec les attributs des classes. Un système de correspondance systématique peut être déduit.

Ainsi, si l'attribut de classe correspondant au nom de la rue où est domicilié un mandataire serait représenté comme suit :

tMdtAdrRueNom

Avec « t » pour indiquer qu'il s'agit d'un champ texte ;  
« Mdt » pour mandataire ;  
« Adr » pour adresse ;  
« Rue » pour rue ;  
« Nom » pour nom de la rue.

On trouvera de la même manière une colonne d'une table

T\_MDT\_ADR\_RUE\_NOM

Ce code doit donc toujours se lire de droite à gauche pour en définir la sémantique => le nom de la rue de l'adresse du mandataire.





Les données complètes de l'adresse d'un mandataire seraient les suivantes :

T\_MDT\_ADR\_RUE\_NOM => le nom de la rue de l'adresse du mandataire  
N\_MDT\_ADR\_RUE\_NUM => le numéro dans la rue de l'adresse du mandataire  
C\_MDT\_ADR\_CPS => le code postal de l'adresse du mandataire  
T\_MDT\_ADR\_LOC => le nom de la localité de l'adresse du mandataire  
T\_MDT\_ADR\_PAY => le nom du pays de l'adresse du mandataire

Le nom du mandataire serait : T\_MDT\_NOM

Cas particulier : les noms d'attributs de type f (devises non €) devront se terminer par une indication du nom de la devise attendue.

### 3.2.5. Liaison table-colonne

Dans la table MANDATAIRE, le nom d'un mandataire est directement dépendant de cette table, ceci se dénote par la syntaxe *Table. Colonne* qui permet d'identifier directement l'origine de la colonne. Cependant cette notion est étroitement liée à l'implémentation physique.

Le formalisme de nommage sera dès lors le suivant :

**Le nom de la donnée doit comporter dans sa dénomination l'acronyme du nom de la classe générique à laquelle il se rapporte.**

Pour la classe **Kmandataire** (acronyme Mdt), on aura vraisemblablement une table **MANDATAIRE**, les données relatives au nom et à l'adresse d'un mandataires reprises dans cette table seront :

Table : MANDATAIRE  
Colonnes : T\_MDT\_NOM  
T\_MDT\_ADR\_RUE\_NOM  
N\_MDT\_ADR\_RUE\_NUM  
C\_MDT\_ADR\_CPS  
T\_MDT\_ADR\_LOC  
T\_MDT\_ADR\_PAY

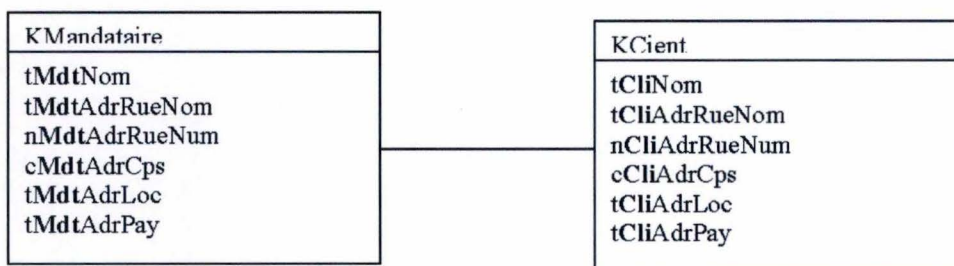
Et donc on écrira MANDATAIRE.T\_MDT\_NOM.

Pour la classe **KsituationJuridique** (acronyme Sitjur), on aura vraisemblablement une table **SITUATION\_JURIDIQUE**, la donnée relative au code de la situation juridique serait la suivante :



### C\_SITJUR\_CODE

Ainsi, si des classes sont liées :



Cela signifie qu'il existe une relation entre un Mandataire et un Client. Cela signifie aussi qu'il y aura dans la table CLIENT une référence à son mandataire.

CLIENT
T_CLI_NOM
T_CLI_ADR_RUE_NOM
N_CLI_ADR_RUE_NUM
C_CLI_ADR_CPS
T_CLI_ADR_LOC
T_CLI_ADR_PAY
T_MDT_NOM

De cette manière, on voit directement que le nom du mandataire ne fait pas partie des données « propres » définissant la classe Kclient. On peut également déduire qu'il vient de la classe dont MDT est l'acronyme c'est-à-dire KMandataire. La traçabilité est ainsi assurée.

De cette manière également, quelle que soit la table où figure la colonne T\_MDT\_NOM, on assure que cette colonne aura toujours la même signification. Ainsi, si pour une raison quelconque (optimisation, dénormalisation, historisation séparée, ...), la table MANDATAIRE venait à devoir être splittée, l'origine de l'information demeurera connue et le lien restera établi vis-à-vis de la donnée originale dans le dictionnaire des données. Là où le lien nom du mandataire – mandataire aurait été perdu si on avait splitté la table MANDATAIRE.T\_NOM en ????.T\_NOM, l'utilisation de l'acronyme de la classe d'origine permet de toujours savoir à quoi se réfère la donnée ; dans le cas d'un split MANDATAIRE.T\_MDT\_NOM deviendra ????.T\_MDT\_NOM ; de même avec la situation juridique SITUATION\_JURIDIQUE.C\_SITJUR\_CODE deviendrait ????.C\_SITJUR\_CODE ce qui reste parlant quant à l'origine de la donnée.



## 12.4. Exemples de la méthodologie FUP

### 12.4.1. Discipline modélisation métier



Service Public Fédéral  
Finances

Support au développement

FUP 0.3

DISCIPLINE –  
MODÉLISATION MÉTIER

#### Statut

document de travail	
version interne	
soumis à acceptation	X
approuvé	

## TABLE DES MATIÈRES

<b>1. PRÉSENTATION DU DOCUMENT.....</b>	<b>3</b>
1.1. INTRODUCTION À LA DISCIPLINE "MODÉLISATION MÉTIER" .....	3
1.2. UML DANS LA DÉMARCHE DE "MODÉLISATION MÉTIER" .....	3
1.3. DANS QUEL ESPRIT ? .....	3
1.4. QUAND DOIT-ON FAIRE DE LA MODÉLISATION MÉTIER ? .....	3
1.5. LES OBJECTIFS DE LA MODÉLISATION MÉTIER AU SPF FINANCES.....	4
1.6. À QUI S'ADRESSE CE DOCUMENT ?.....	4
1.7. LES CONVENTIONS DANS CE DOCUMENT .....	5
<b>2. POSITIONNEMENT PAR RAPPORT AU CYCLE DE VIE.....</b>	<b>5</b>
<b>3. DESCRIPTION SCHÉMATIQUE DE LA DÉMARCHE.....</b>	<b>6</b>
3.1. LES ENTRÉES .....	6
3.2. LES SORTIES .....	7
3.3. L'ENCHAÎNEMENT DES ACTIVITÉS .....	8
3.4. LES ARTÉFACTS PRODUITS.....	8
<b>4. DESCRIPTION DES ACTIVITÉS.....</b>	<b>9</b>
4.1. IDENTIFIER LES ACTEURS ET PROCESSUS MÉTIER (A1) .....	9
4.1.1. Identifier les acteurs.....	9
4.1.2. Identifier les processus métier du domaine étudié .....	10
4.2. DÉCRIRE LES PROCESSUS MÉTIER (A2) .....	11
4.2.1. Décrire textuellement les processus métier .....	11
4.2.2. Décrire graphiquement les processus métier .....	12
4.3. MODÉLISER LES ACTEURS MÉTIER (A3) .....	13
4.3.1. Identifier les relations entre les acteurs métier.....	13
4.3.2. Décrire les Acteurs Métier.....	14
4.4. IDENTIFIER ET DÉCRIRE LES ENTITÉS MÉTIER (A4) .....	14
4.5. MODÉLISER LES ENTITÉS MÉTIER (A5).....	16
4.5.1. Regrouper les entités métier en packages .....	16
4.5.2. Décrire le cycle de vie des entités .....	17
4.6. PRODUIRE LE GLOSSAIRE (A6) .....	18
<b>5. RESPONSABILITÉS – VUE PAR RÔLE.....</b>	<b>19</b>
<b>6. ANNEXES.....</b>	<b>20</b>
6.1. DOCUMENTS UTILES .....	20
6.1.1. Modèles ou document-type .....	20
6.1.2. Procédures associées et guides .....	20

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	2/20



## 1. PRESENTATION DU DOCUMENT

### **1.1. Introduction à la discipline "Modélisation Métier"**

Ce document présente la démarche de modélisation métier telle que préconisée dans le cadre de FUP.

Le document propose un canevas pour réaliser « l'analyse du métier » du processus que l'on vise à informatiser. Il est destiné à servir de support aux analystes fonctionnels du SPF Finances.

La démarche présentée est alignée sur les concepts développée par UP dans la discipline *Business Modeling*.

### **1.2. UML dans la démarche de "Modélisation Métier"**

L'application de la démarche ne nécessite pas la connaissance de l'ensemble des diagrammes UML, seul un sous-ensemble limité de diagrammes est exploité dans cette discipline.

Nous préconisons d'utiliser UML pour favoriser la convergence et la cohérence entre les concepts qui sont modélisés dans cette discipline et ceux modélisés dans les autres disciplines de la méthode : Exigences, Analyse et Conception.

### **1.3. Dans quel esprit ?**

Dans cette discipline, on évitera de parler « informatique », on cherchera surtout à décrire les processus en faisant abstraction de comment ils peuvent ou pourraient être implémentés par un système d'information et éventuellement ou probablement par un système informatique.

### **1.4. Quand doit-on faire de la modélisation métier ?**

La discipline **modélisation métier** est une **discipline amont optionnelle** par rapport aux cinq autres disciplines fondamentales de FUP qui sont l'expression des besoins / exigences, l'analyse et la conception, l'implémentation, le test et le déploiement.

Elle est exécutée en phase d'*Inception*.

Les documents du BPR Coperfin sont certainement des documents intéressants pour initier cette analyse.

Elle sera **optionnelle** si les processus métiers à analyser sont déjà décrits dans un document. Néanmoins si la notation utilisée dans ce document n'est pas standard, on pourra éventuellement retranscrire les processus qui nous intéressent à l'aide de diagrammes UML, sur base des informations laissées par des travaux précédents, tel que Coperfin.

Par contre si on a affaire à de nouveaux processus, on pourra tout à fait envisager de consacrer une partie de la phase d'*Inception* à modéliser ces processus, pour ensuite en extraire les cas d'utilisations et poursuivre avec la discipline expression des besoins / exigences sur les cas d'utilisations extraits.

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	3/20

### 1.5. Les objectifs de la modélisation métier au SPF Finances

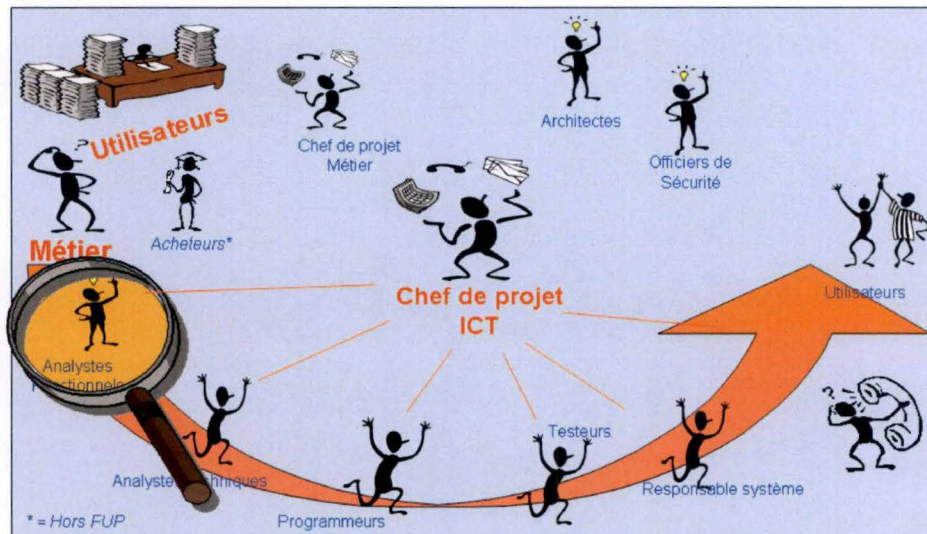
- ☞ L'objectif de l'analyse du métier est de comprendre, de documenter et d'expliquer les activités et les processus métiers du SPF Finances.
- ☞ A partir du modèle produit, les analystes fonctionnels pourront identifier les activités métiers qu'un projet se propose d'informatiser au travers d'une application (/qui seront informatisées par une application). Ces activités seront ensuite détaillées et décrites dans la discipline expression des besoins / exigences.
- ☞ Les modèles produits contribueront à alimenter la pré-étude (Voir le document Cycle de vie)

*Formaliser le métier  
en terme de  
processus,  
d'événements,  
d'entités métiers,  
d'acteurs métiers et  
de collaborateurs.*

*Préparer ou  
Alimenter  
la pré-étude*

### 1.6. A qui s'adresse ce document ?


Il s'adresse aux **analystes fonctionnels** du SPF Finances qui détiennent une connaissance d'un domaine métier, et qui partent d'une première expression de besoins rédigée par des utilisateurs.



<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	4/20



### 1.7. Les conventions dans ce document


Dans ce document, les éléments de modélisation produits *non UML* (généralement des documents ou des descriptions textuelles) à l'issue de chacune des activités de la démarche seront identifiés par le sigle .

Par exemple :

 Description textuelle de processus métier

Les éléments de modélisation produits *en UML* à l'issue de chacune des activités de la démarche seront identifiés par le sigle .

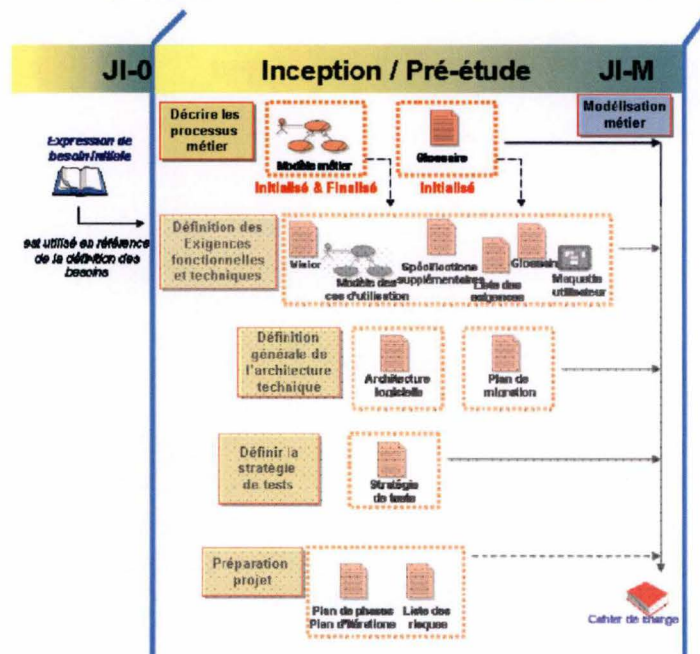
Par exemple :

 Diagramme d'activités

## 2. POSITIONNEMENT PAR RAPPORT AU CYCLE DE VIE

L'analyse métier est conduite en amont des projets, durant la phase d'Inception entre les Jalons Inc-0 et Inc-n.

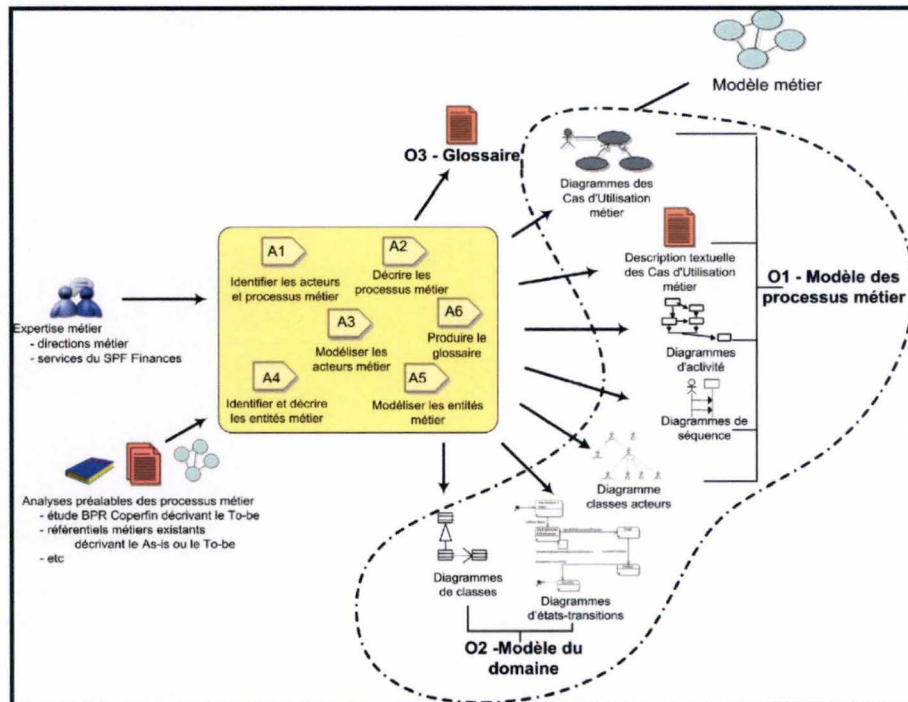
La discipline Modélisation métier s'effectue essentiellement dans la phase d'Inception et les artefacts produits pourront servir à la discipline Expression des besoins / Exigences. Ils pourront également servir à confectionner le cahier de charge s'il y a besoin.



Préparé par	Nom et localisation de fichier	Date/Version	Page
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	5/20

### 3. DESCRIPTION SCHEMATIQUE DE LA DEMARCHE

La figure suivante montre les entrées, les sorties, et les activités de la discipline *Modélisation Métier*.



#### 3.1. Les entrées

En entrée, les analystes fonctionnels pourront s'appuyer sur :

- ↳ L'expertise métier issue des directions métiers, et des différents services du SPF Finances.
- ↳ Des analyses préalables des processus métier : étude BPR Coperfin décrivant le To-be, référentiels métiers existants, modèles métier existant décrivant soit le As-is soit le To-be, etc.

Les modèles en entrée peuvent donc avoir été réalisés avec un autre formalisme que celui préconisé par FUP.

Préparé par	Nom et localisation de fichier	Date/Version	Page
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	6/20



### 3.2. Les sorties

Les différents artefacts produits par l'analyse métier sont :

↳ Le **Modèle métier**, qui consiste en 2 parties :

○ Le Modèle des processus métier (O1)

Les besoins métiers sont cernés et communiqués à travers les cas d'utilisation métier. Les cas d'utilisation métier seront le point de départ pour l'élaboration des modèles de cas d'utilisation système dans la discipline *Expression des besoins / Exigences*.

Ce modèle est composé :

- du diagramme des classes acteurs
- des diagrammes de cas d'utilisation métier
- des descriptions textuelles des cas d'utilisation métier
- des diagrammes d'activité
- des diagrammes de séquence (facultatif)

○ Le Modèle du domaine (O2) ou « Modèle des entités métier »

Le modèle des entités métier conduira à identifier les classes d'analyse dans la discipline *Analyse et Conception*. Il est composé :

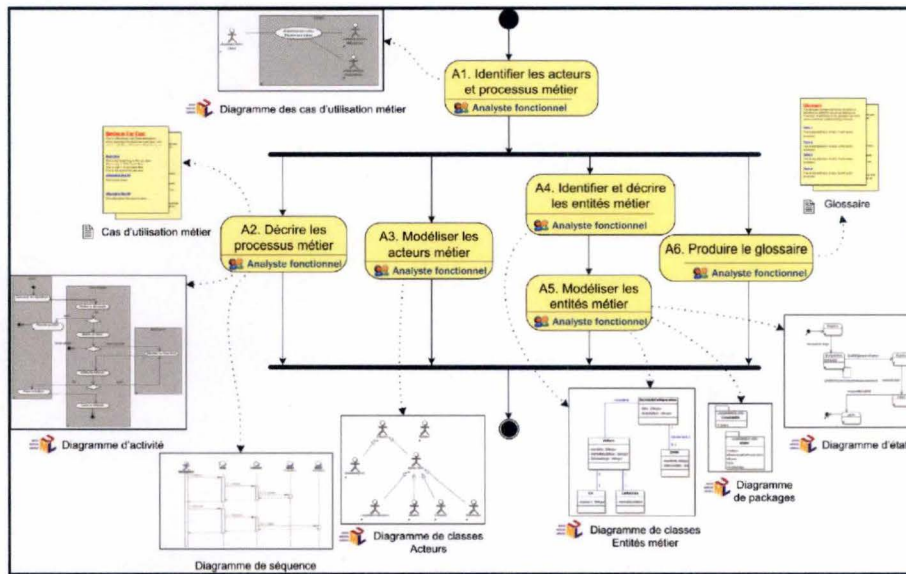
- d'un diagramme de classes modélisant les entités métier
- des diagrammes d'états – transitions pour certaines entités métiers

↳ Le **Glossaire (O3)**, qui est transverse à tous les projets et à tous les processus métier.

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	7/20

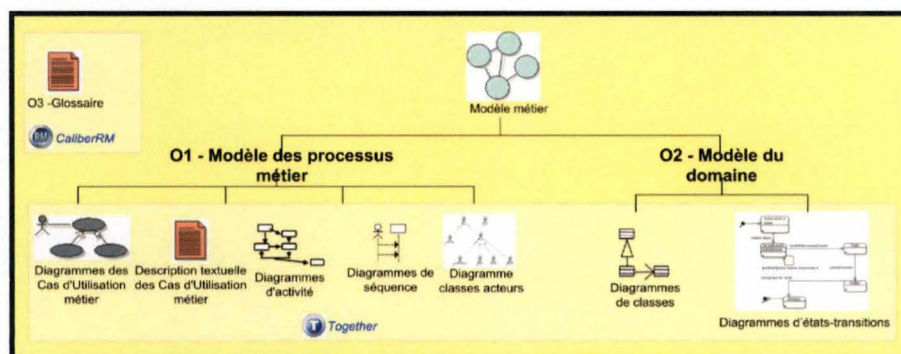
### 3.3. L'enchaînement des activités

Le diagramme suivant montre l'enchaînement des activités entreprises dans le cadre de la discipline *Modélisation métier*, ces activités sont décrites et détaillées dans le chapitre 4 – Description des activités :



### 3.4. Les artefacts produits

Le schéma ci-dessous montre les différents artefacts produits par les activités de la discipline, dans leur hiérarchie :



Préparé par	Nom et localisation de fichier	Date/Version	Page
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	8/20



## 4. DESCRIPTION DES ACTIVITES

Note : Les activités suivantes ne sont pas décrites dans un ordre chronologique ou séquentiel car elles sont exécutées tout au long de la discipline.

### 4.1. Identifier les acteurs et processus métier (A1)

A ce niveau, on va **identifier** les acteurs et processus métier qui participent et collaborent au domaine que l'on étudie.

Dans cette activité, on utilise le diagramme des cas d'utilisation UML afin de représenter ces acteurs, ces processus et les relations qui les lient.

#### 4.1.1. Identifier les acteurs

##### Objectif

Pour démarrer, on identifie les acteurs métiers qui participent et collaborent aux processus étudiés. Les acteurs représentent des rôles ou des personnes qui participent aux processus métiers.

##### Démarche

Les acteurs que l'on identifie peuvent éventuellement être catégorisés en :

- ✚ **Business Actor**, qui sont **externes** au domaine étudié (attention, il s'agit bien d'une notion interne - externe par rapport au domaine étudié, pas par rapport au SPF Finances ! Il n'y a pas dans cette catégorisation de notion de localisation physique. Ainsi, un acteur tout à fait interne au SPF Finances pourrait être Business Actor dans un certain domaine...)
- ✚ **Business Worker**, qui sont **internes** au domaine étudié; on pourra éventuellement les classer en 2 sous-types :
  - les « Case worker », est en contact avec l'extérieur du domaine, c'est-à-dire avec les Business Actors.
  - les « Internal worker » n'ont aucun contact avec l'extérieur, ils n'ont donc aucune interaction avec les Business Actors.

Les acteurs métier sont représentés par des *Actors UML*, éventuellement stéréotypés <<Business Actor>> ou <<Business Worker>>.

**Notez bien** que ce qui est important, c'est l'identification et la description des acteurs, leur stéréotypage est tout à fait facultatif.

L'ensemble de ces acteurs va former le dictionnaire des acteurs et collaborateurs métier.

##### Rôle responsable

Analyste fonctionnel

##### Artéfacts produits

- ✚  Acteurs, éventuellement stéréotypés <<Business Actor>> ou <<Business Worker>>.

Exemple :

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	9/20



#### 4.1.2. Identifier les processus métier du domaine étudié

##### Objectifs

Le but de cette activité est d'identifier les processus métier pour le domaine que l'on étudie.

##### Démarche

Pour identifier les processus métier, on peut procéder de deux manières :

- ✎ On peut commencer par identifier les macro-processus que l'on décomposera ensuite jusqu'aux activités, c'est une démarche de type top-down (c'est souvent adapté pour un nouveau processus).
- ✎ On peut aussi d'abord commencer par identifier les acteurs et les collaborateurs et on cherche à savoir « ce qu'ils font », ce sont les activités, ensuite on regroupe ces activités dans des processus de plus haut niveau, c'est une démarche de type bottom-up (souvent adapté à la description d'un processus existant)

On peut aussi ensuite chercher à identifier des dépendances entre les processus que l'on a identifiés et représentés sur le diagramme de cas d'utilisation UML.

Chaque processus identifié doit être illustré à l'aide d'un diagramme de cas d'utilisation UML comme on le fait pour les use case systèmes.

Nous considérerons les sous-processus comme des activités, et nous les représentons à l'aide de diagrammes d'activités UML.

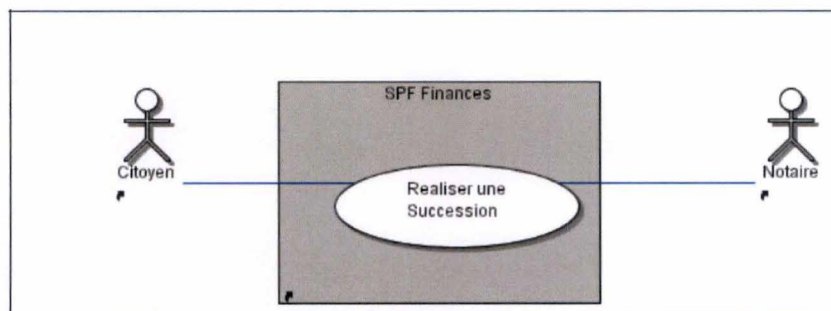
##### Rôle responsable

Analyste fonctionnel

##### Artéfact produit

- ✎  Diagramme de Cas d'Utilisation.

##### Exemple :



<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	10/20



## 4.2. Décrire les processus métier (A2)

A ce niveau, on va **décrire** les **processus métiers** identifiés précédemment.

Cette activité est réalisée en parallèle de l'activité d'identification des processus métiers et acteurs métier.

La description d'un processus métier est l'élément le plus important de sa modélisation.

Pour décrire un processus métier, on produit une description textuelle, qui sera tout ou partie formalisée graphiquement par un diagramme d'activité UML et/ou un ou des diagrammes de séquence UML.

Dans cette activité on utilisera donc :

- ✎ le **modèle UC** pour décrire textuellement le processus,
- ✎ le **diagramme d'activité** pour le décrire graphiquement,
- ✎ un ou des **diagrammes de séquence** pour décrire graphiquement un ou des scénarii du processus (si des alternatives existent, par exemple).

### 4.2.1. Décrire textuellement les processus métier

#### Objectif

Pour chaque processus que l'on a identifié dans le modèle des cas d'utilisation métier, on va d'abord le décrire à l'aide d'un **plan-type** c'est-à-dire un texte structuré de la même manière pour tous les processus étudiés, appelé "Spécification du Cas d'Utilisation Métier"<sup>1</sup>.

#### Démarche

##### 1. Description textuelle (business UC specification)

###### 1.1 Identification

###### 1.2 Résumé

###### 1.3 Acteurs métier

###### 1.4 Collaborateurs métier

###### 1.5 Événements métier reçus

###### 1.6 Pré conditions

###### 1.7 Workflows

###### 1.7.1 Scénario nominal

###### 1.7.2 Scénarii alternatifs

1-a : Scénario alternatif a de l'étape 1

1-b : Scénario alternatif b de l'étape 1

2-a : Scénario alternatif a de l'étape 2

###### 1.8 Post conditions

###### 1.9 Événements métier produits

###### 1.10 Règles de gestion

Spécification du cas d'utilisation métier


- ✎ Résumer l'**objectif** du processus métier,
- ✎ Décrire les **acteurs métiers externes** qui participent au processus (Acteurs stéréotypés "Business Actor"),
- ✎ Identifier et décrire les **collaborateurs métiers internes** intervenant dans son exécution et contribuant à la réalisation du processus (Acteurs stéréotypés "Case worker" et "Internal worker"),
- ✎ Identifier les **événements métier reçus** ou **déclencheurs** du processus,
- ✎ Décrire les **pré-conditions** : on décrit ici les conditions qui doivent être satisfaites pour pouvoir déclencher le processus,
- ✎ Décrire le **processus lui-même**, sa dynamique : ici on décrit en langage naturel le déroulement nominal du processus (en identifiant les activités métier qui le composent) et les principales alternatives si elles existent comme on le ferait pour un cas d'utilisation normal (cas d'utilisation système d'une application informatique),
- ✎ Décrire les **post-conditions** : ici on va décrire les conditions qui doivent être satisfaites lorsque le processus est terminé,
- ✎ Décrire les **résultats** qu'il produit (événements métier produits ),
- ✎ Identifier et décrire les **règles de gestion** : ici on va décrire les règles de gestion métier qui s'appliquent lors du déroulement du processus métier, et on va donner à ces règles de gestion une définition précise.

<sup>1</sup> Artéfact de FUP dont un modèle est donné dans l'ensemble des modèles d'artéfacts. Il est repris dans le livrable "Dossier d'analyse métier".

Préparé par	Nom et localisation de fichier	Date/Version	Page
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	11/20

**Rôle responsable**

Analyste fonctionnel

**Artéfact produit**
 Cas d'utilisation métier
**4.2.2. Décrire graphiquement les processus métier****Objectif**

Lorsqu'on décrit textuellement un processus métier, il est souvent utile de s'appuyer sur des diagrammes qui permettent de mieux comprendre les aspects dynamiques. Pour cela on utilisera le diagramme d'activité UML qui va permettre de :

- ↳ Mieux décrire la logique de séquence ment des activités métier qui composent le processus métier et la cinématique.
- ↳ Faire apparaître les événements métier reçus ou produits par ce processus.
- ↳ Faire apparaître éventuellement certaines entités manipulées par ce processus, et si ces entités sont utilisées ou produites par le processus. On pourra également préciser l'état de ces entités lorsqu'elles sont impactés ou interviennent au niveau des activités.
- ↳ Faire mieux apparaître quel collaborateur métier réalise quelle activité du processus.

Pour les processus comportant un flux complexe, il peut être nécessaire de décrire graphiquement un ou plusieurs scénarii (parcours d'un flux du processus) à l'aide de diagrammes de séquence UML.

Notez que ces diagrammes (collaboration ou séquence) sont optionnels.

**Démarche**

On représentera sur le diagramme d'activité les acteurs et les collaborateurs qui participent ou réalisent le processus par des "couloirs de responsabilité" ou des swimlanes, et on placera dans ces couloirs les activités dont ils sont responsables.

Pour les processus métiers présentant un flux complexe, on sélectionnera les scénarii les plus importants (ou tous, si nécessaire) et l'on représentera chaque scénario dans un diagramme de séquence UML.

**Rôle responsable**

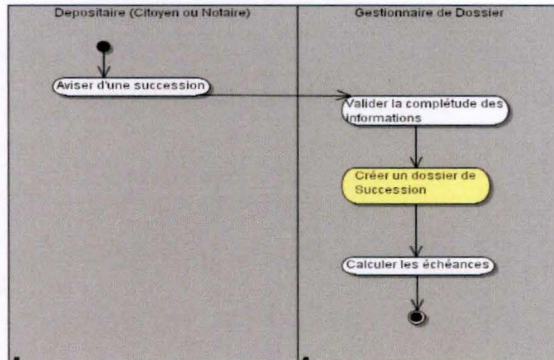
Analyste fonctionnel

**Artéfact produit**
 Diagramme d'activité

 Diagramme de séquence
Exemple de diagramme d'activité

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	12/20





### 4.3. Modéliser les acteurs métier (A3)

#### 4.3.1. Identifier les relations entre les acteurs métier

##### Objectif

Une fois les acteurs identifiés, il est possible d'identifier et représenter les relations entre ces acteurs. La représentation de l'ensemble des acteurs et leurs relations peut être faite au moyen d'un diagramme de classes UML.

##### Rôle responsable

Analyste fonctionnel

##### Artéfact produit

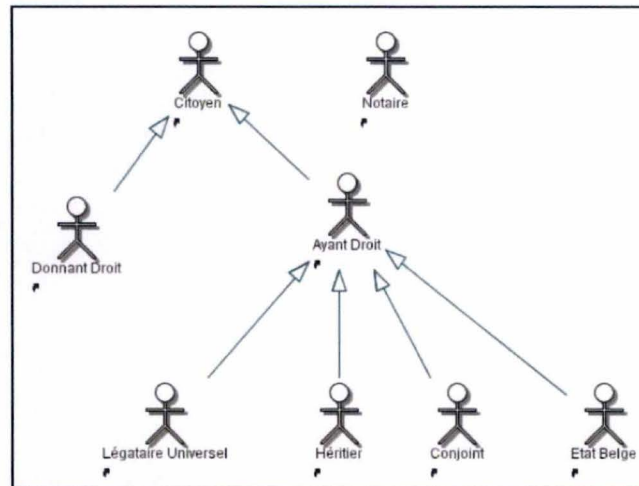
  Diagramme de classes – Acteurs métier

##### Exemple :

Dans l'exemple ci-dessous, nous avons identifié pour le processus de successions deux types d'acteurs importants qui sont l'Ayant Droit et le Donnant Droit, ces acteurs sont tous les deux des Citoyens, c'est ce que l'on a représenté sur le diagramme.

L'acteur « Citoyen » représente un acteur « généralisé », qui permet de regrouper, généraliser les acteurs « Donnant Droit » et « Ayant Droit » en un rôle commun.

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	13/20



#### 4.3.2. Décrire les Acteurs Métier

##### Objectif

Le but de cette action est de décrire dans le dictionnaire des acteurs et collaborateurs métier chaque acteur métier.

##### Démarche

Chaque acteur métier est passé en revue, et sa description est ajoutée au dictionnaire des acteurs et collaborateurs métier.

##### Rôle responsable

Analyste fonctionnel

#### **4.4. Identifier et décrire les entités métier (A4)**

A ce niveau, on va **identifier et décrire** les entités métiers du domaine que l'on étudie.

Un bon moyen de vérifier que le modèle produit est cohérent est de demander aux experts métier de le valider, ils devront pouvoir retrouver dans le modèle tout le vocabulaire de leur métier.

Attention, les concepts liés à la mise en oeuvre du système informatique ne doivent pas apparaître à ce niveau, il ne s'agit pas de décrire le modèle de données d'une application mais de décrire des objets métier de haut niveau : une succession, etc.

Cette activité est réalisée en parallèle de l'activité de description des processus métier.

En effet, c'est en décrivant les processus métier, que l'on va pouvoir « découvrir » les entités métiers manipulées par le processus, et dans certains cas

On va **représenter les entités métier** et les relations entre ces entités à l'aide du **diagramme de classes UML**.

##### Objectif

A ce stade de la modélisation, on va identifier toutes les entités métiers qui contribuent à la réalisation des processus du domaine modélisé. Ces entités sont consommées, transformées ou produites par les processus.

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	14/20



Ensuite on va chercher à décrire leurs attributs et enfin on identifiera les relations entre ces entités.

Les concepts seront modélisés dans un diagramme de classes UML.

### Démarche

On veillera lors de l'identification des entités métier à réutiliser au maximum les entités métier déjà identifiées dans le cadre d'autres travaux de modélisation qui auraient pu déjà avoir lieu sur d'autres projets.

L'idéal étant de maintenir un seul référentiel global des entités métier, ainsi que des processus métier par ailleurs.

A ce stade, l'identification des entités que l'on va réaliser ici ne sera pas figée, elle pourra encore évoluer (modifications et enrichissements) dans les actions suivantes.

On identifie les entités métiers à partir de l'ensemble des descriptions textuelles des processus que l'on a réalisés à l'étape précédente. Souvent par exemple, on peut facilement les repérer dans les substantifs et les adjectifs, ce sont de bons candidats à devenir des entités ou des attributs d'entités.

### Rôle responsable

Analyste fonctionnel

### Artéfact produit

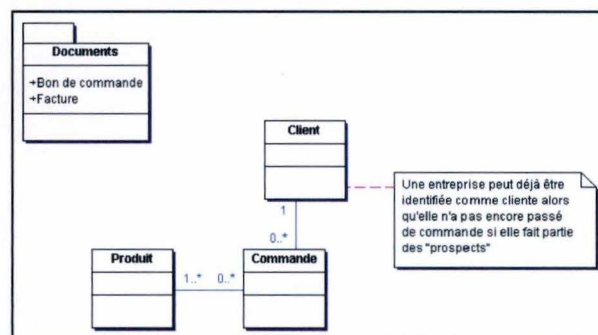
Cette identification produit un ou plusieurs diagrammes de classes.

Le modèle produit s'appelle le Business Object Model. Lorsque le diagramme ne contient que les entités (sans les acteurs), cela s'appelle un **modèle du domaine**.



Diagramme de classes

Exemple :



<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	15/20

#### **4.5. Modéliser les entités métier (A5)**

Après avoir identifié et décrit les entités métier, on va d'une part structurer ces entités en les regroupant par packages et d'autre part, lorsque cela est significatif, **décrire le cycle de vie de ces entités**, c'est-à-dire les différents états dans lesquels elle va pouvoir se trouver, on utilisera pour cela les **diagrammes d'états UML**.

En résumé, dans cette activité on utilisera :

- ↳ le **diagramme de classes** déjà initié lors de l'identification et description des entités métier, il sera complété de la notion de package;
- ↳ le **diagramme d'états** pour décrire le cycle de vie des entités.

##### **4.5.1. Regrouper les entités métier en packages**

###### **Objectif**

Dans cette activité, on cherchera à **structurer le modèle des entités métiers** produit à l'étape précédente.

Pour cela, on identifiera les regroupements logiques d'entités métier, afin d'organiser le domaine métier considéré et d'en avoir une vue à un plus haut niveau d'abstraction.

###### **Démarche**

Pour structurer le modèle des entités, il y a plusieurs solutions. Le choix de la solution dépend fortement du domaine métier analysé :


- ↳ Soit on cherche à identifier des catégories d'entités,
- ↳ Soit on fera un regroupement par domaine d'activités (comptabilité, facturation, ...) ou on cherchera à mapper les entités sur l'organisation du SPF Finances, etc.


###### **Rôle responsable**

Analyste fonctionnel

###### **Artéfacts produits**

Cette structuration impacte les diagrammes de classes produits à l'étape précédente, et est également documentée au niveau d'un ou plusieurs diagrammes de packages.

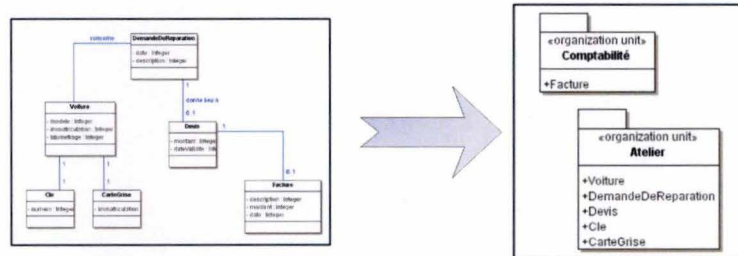
↳  Diagramme de classes

↳  Diagramme de packages

Exemple :

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	16/20





#### 4.5.2. Décrire le cycle de vie des entités

##### Objectif

L'étape suivante dans la modélisation des entités métier consiste à identifier, lorsque cela est pertinent, l'ensemble des états dans lequel peut se trouver une entité, ainsi que les transitions entre ces états.

C'est ce qu'on appelle le cycle de vie d'une entité.

##### Démarche

L'identification des états d'une entité métier peut être faite en s'appuyant sur les descriptions textuelles de processus métier. Il faut essayer d'être exhaustif lorsqu'on réalise ce type de diagramme, on cherche à recenser l'ensemble des états et des transitions.

##### Rôle responsable

Analyste fonctionnel

##### Artéfact produit

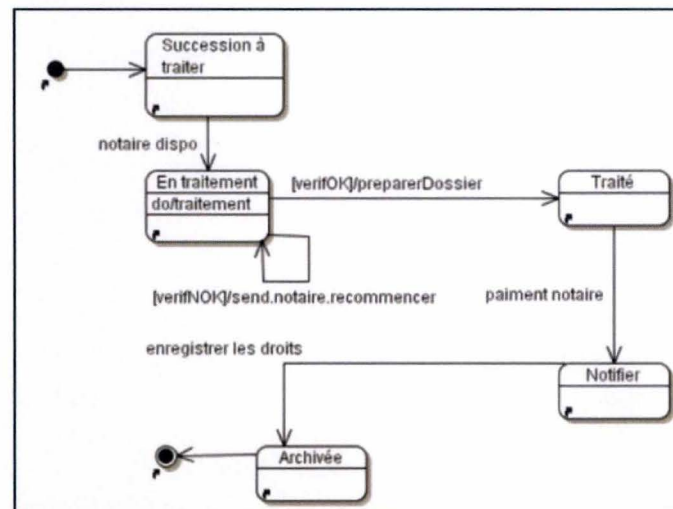
Pour décrire le cycle de vie des entités métiers, on utilisera le diagramme d'états – transitions :



Diagramme d'états – transitions

##### Exemple :

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	17/20



NB : La description du cycle de vie des entités est optionnelle.

#### 4.6. Produire le glossaire (A6)

##### Objectif

Cette activité consiste à déterminer le vocabulaire métier pour les termes du domaine métier analysé. Il permet d'éviter les ambiguïtés et les interprétations erronées.

Ce glossaire sera initié, raffiné et enrichi tout au long des phases d'inception et d'élaboration.

Il est commun à tout le SPF Finances, pour jouer son rôle d'intégrateur de concepts.

##### Démarche

Chaque entrée du glossaire métier sera soumise au gestionnaire du glossaire et comportera un nom de terme ainsi que sa définition (un texte), le tout dans les 2 langues. Le gestionnaire, en collaboration avec les administrations et projets concernés, vérifiera la définition donnée, la pertinence du terme ainsi que sa non existence dans le glossaire sous une autre appellation.

##### Rôle responsable

Analyste fonctionnel

##### Artéfact produit

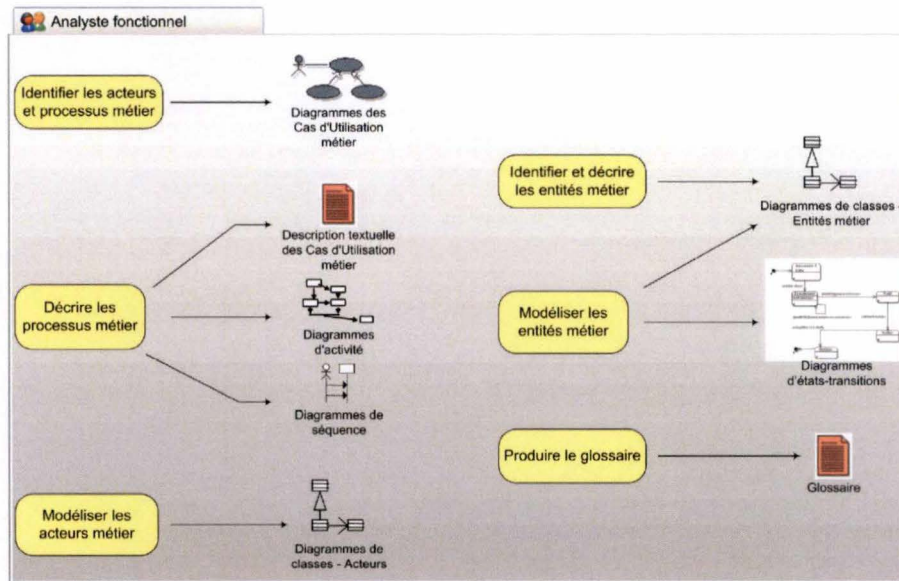
📄 Glossaire

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	18/20



## 5. RESPONSABILITES – VUE PAR ROLE

Le schéma ci-dessous montre une vue synthétique des rôles impliqués, activités et artefacts produits au sein de la discipline :



<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	19/20

## 6. ANNEXES

### 6.1. Documents utiles

#### 6.1.1. Modèles ou document-type

Cette section reprend la liste des modèles d'artefacts à utiliser dans la démarche de cette discipline. Pour rappel, les modèles sont compris dans FUP et font l'objet de documents à part entière.

- ↳ Modèle de spécification du Cas d'utilisation Métier (description de processus métier).
- ↳ Modèle de dossier d'analyse métier. Il s'agit du livrable regroupant entre autres les spécifications des différents cas d'utilisation métier.

#### 6.1.2. Procédures associées et guides

Le mode opératoire de la discipline *Modélisation Métier*, indiquant pour tout ce qui a été exposé dans le présent document comment le réaliser avec les outils de SupDev, est repris dans le document suivant :

- ↳ Mode opératoire de la discipline Modélisation métier

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-Discipline_Modelisation_Metier v0[1].8	5/26/2006/ 0.8	20/20



## 12.4.2. Discipline modélisation métier – Mode opératoire – Présentation générale



Service Public Fédéral  
Finances

Support au développement

# FUP 0.3

## DISCIPLINE

## MODELISATION METIER

## MODE OPÉRATOIRE

### Statut

document de travail	
version interne	
soumis à acceptation	X
approuvé	

## TABLE DES MATIÈRES

<b>1. INTRODUCTION .....</b>	<b>3</b>
1.1. OBJET DU DOCUMENT .....	3
1.2. ARTEFACTS À PRODUIRE .....	3
<b>2. STRUCTURE DES ARTEFACTS DANS CALIBRM.....</b>	<b>4</b>
2.1. RÈGLES BUSINESS (O1) .....	4
2.2. GLOSSAIRE MÉTIER (O3) .....	5
<b>3. STRUCTURE DES ARTEFACTS DANS TOGETHER.....</b>	<b>6</b>
3.1. PRÉREQUIS .....	6
3.2. MODÉLISATION MÉTIER .....	7
3.3. ACTEURS MÉTIERS (O1) .....	8
3.4. PROCESSUS MÉTIER (O1) .....	9
3.5. MODÈLE DU DOMAINE (O2) .....	10
<b>4. RÉCAPITULATIF DES MODES OPÉRATOIRES DÉTAILLÉS.....</b>	<b>11</b>

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-Discipline Modélisation Métier_fr v0[1].1	5/26/2006/ 0.1	2/11



## 1. INTRODUCTION

### 1.1. Objet du document

Le but de ce document est de synthétiser les artefacts à produire dans la discipline 'Modélisation métier' et de référencer les outils et structures nécessaires à cette production.

Ce document constitue le point d'entrée des modes opératoires détaillés.

### 1.2. Artefacts à produire

Modèle des acteurs et processus métier (O1)

Artefacts	Together	CalibrM
Diagramme classes acteurs	Dictionnaire acteurs métier	
Modèle des processus	BP00-Modèle des processus	
<i>Par Processus</i>	<i>BPxx-Nom process-GRP</i>	
• Description textuelle	BPxx-Nom process-UD	Règle business
• Diagramme d'activité	BPxx-Nom process-AD	
• Diagramme de séquence	BPxx-Nom process-SD	

Modèle du domaine (O2)

Artefacts	Together	CalibrM
Diagramme des classes	Modèle entités métier	
<i>Par entité</i>		
• Diagramme d'état	<i>Nom entité-SC</i>	

Le glossaire métier (O3)

Artefacts	Together	CalibrM
Acronymes		Requirement ACRO
Glossaire des concepts		Requirement GLOSS

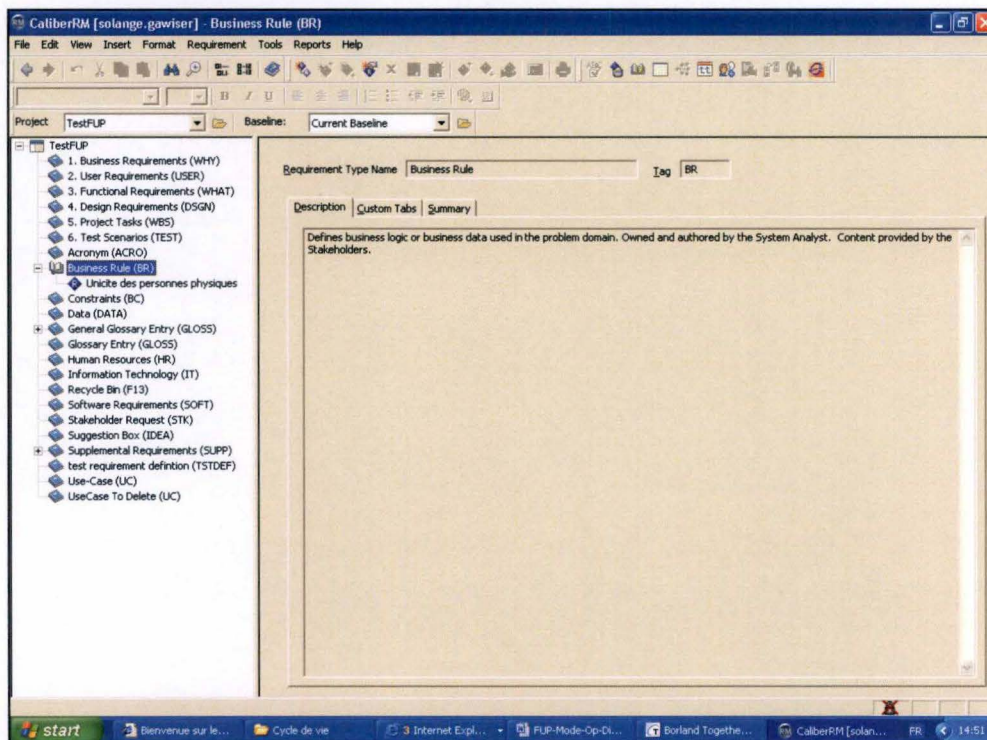
Préparé par	Nom et localisation de fichier	Date/Version	Page
	FUP-MO-Discipline Modélisation Métier_fr v0[1].1	5/26/2006/ 0.1	3/11

## 2. STRUCTURE DES ARTEFACTS DANS CALIBRRM

### 2.1. Règles business (O1)

Les règles métier sont introduites par projet, au niveau de la rubrique 'Business Rule'

Mode opératoire détaillé : FUP-MO-D1-Règles Business



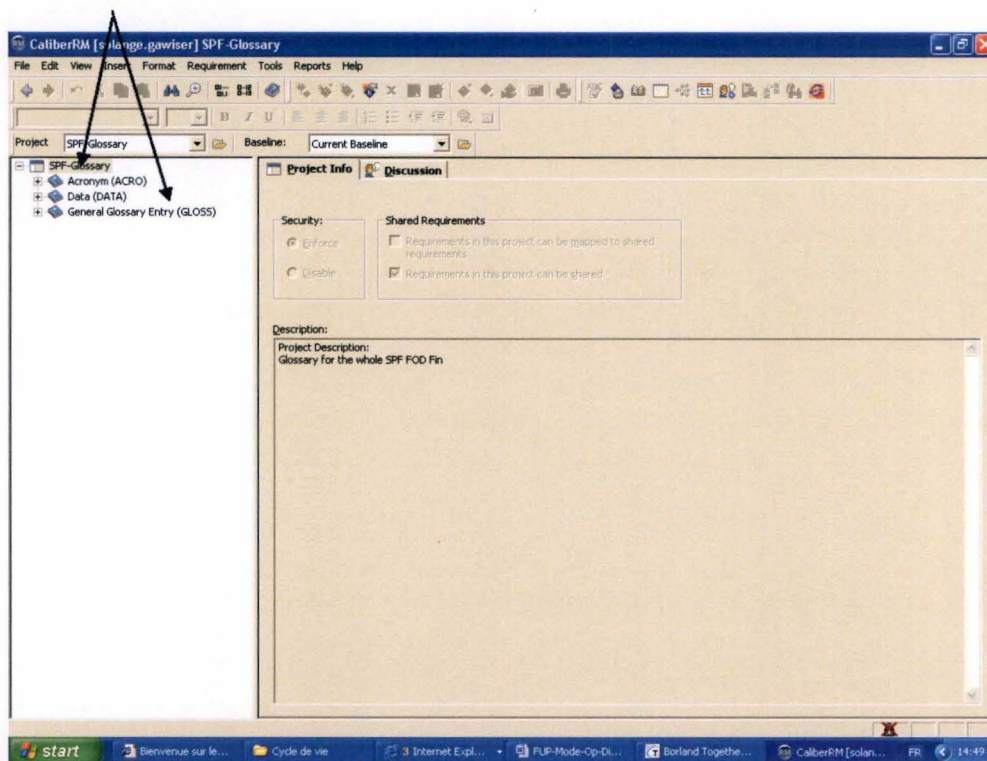
<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO- Discipline Modélisation Métier_fr v0[1].1	5/26/2006/ 0.1	4/11



## 2.2. Glossaire Métier (O3)

Le glossaire métier est géré au niveau global SPF. Il centralise pour tous les projets, d'une part les acronymes utilisés pour les conventions de nommage et d'autre part la définition de tous les termes et concepts métier.

Mode opératoire détaillé : FUP-MO-D1-Glossaire

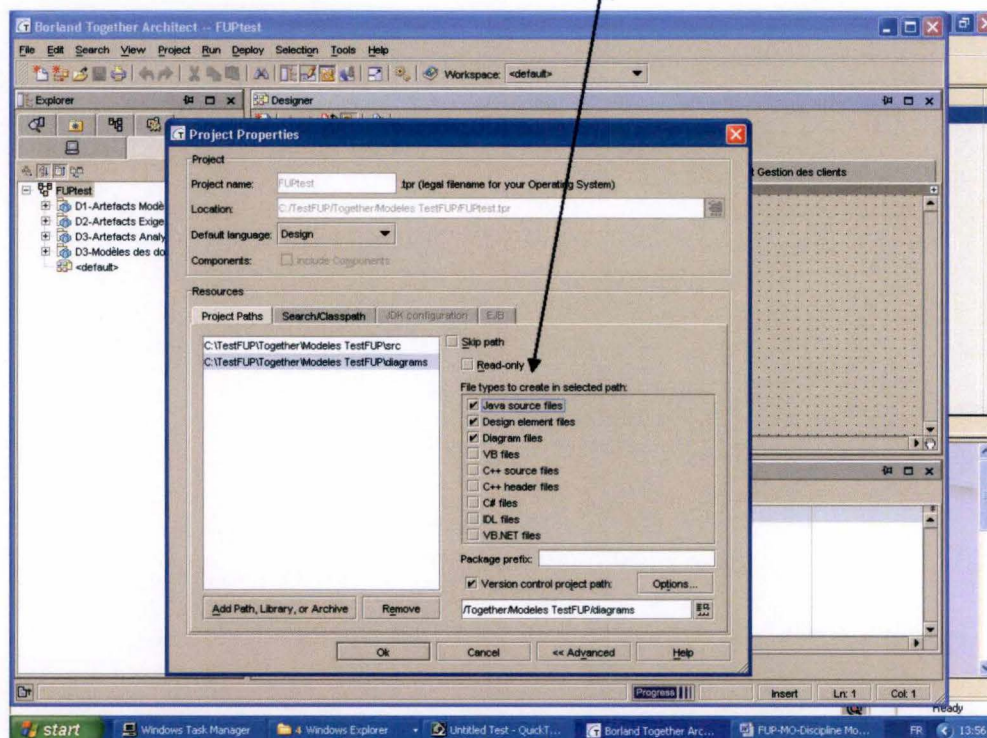


<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-Discipline Modélisation Métier_fr v0[1].1	5/26/2006/ 0.1	5/11

### 3. STRUCTURE DES ARTEFACTS DANS TOGETHER

#### 3.1. Prérequis

Vérifier que votre projet dans Together a bien été initialisé avec l'option 'création fichiers Java', ce qui aura comme impact de devoir respecter les standards Java en ce qui concerne la nomenclature de tous les objets (Nom des packages, diagrammes, classes,...). Tous les blancs dans les noms mentionnés dans les modes opératoires doivent donc être encodés obligatoirement à l'aide du caractère '\_' dans Together.

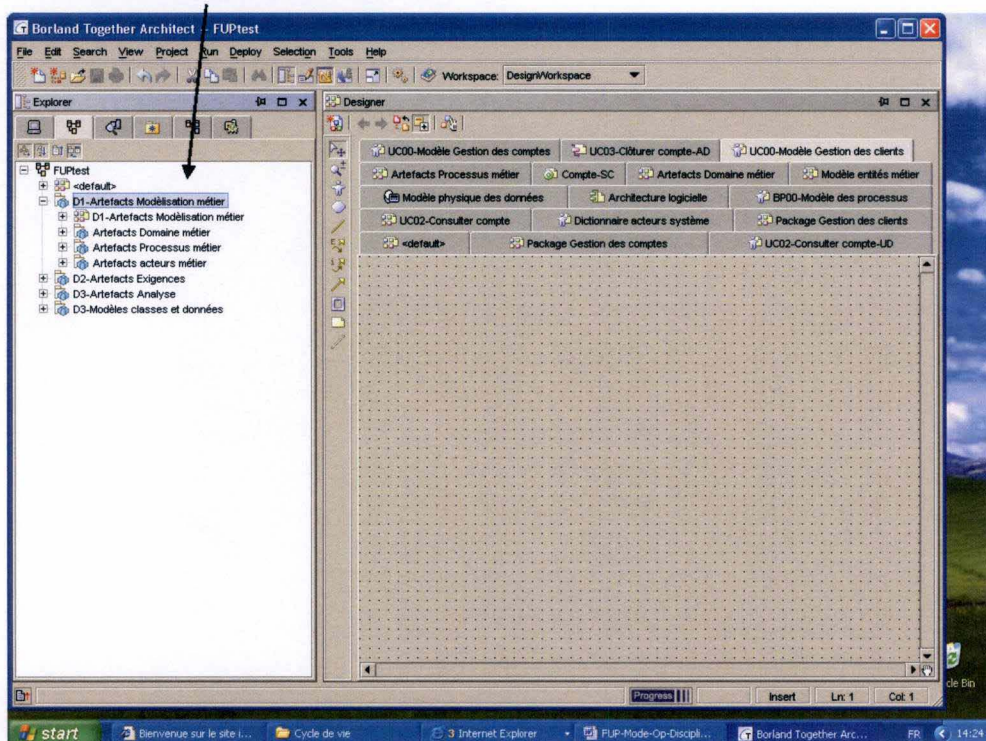


Préparé par	Nom et localisation de fichier	Date/Version	Page
	FUP-MO-Discipline Modélisation Mýýtier_fr v0[1].1	5/26/2006/ 0.1	6/11



### 3.2. Modélisation métier

Le Package 'D1-Artéfacts Modélisation métier' contient tous les éléments de modélisation gérés au niveau de la discipline modélisation métier

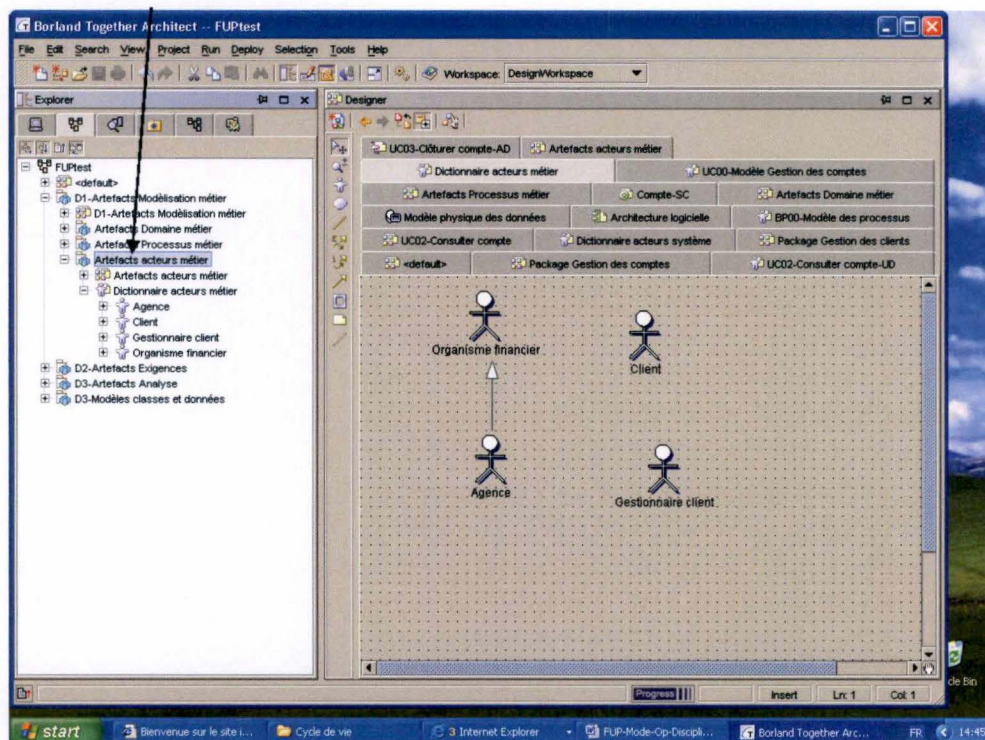


<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-Discipline Modélisation Mýýtier_fr v0[1].1	5/26/2006/ 0.1	7/11

### 3.3. Acteurs métiers (O1)

Le Package 'Artéfacts Acteurs métier' contient tous les éléments liés à la modélisation des acteurs métier.

Mode opératoire détaillé : FUP-MO-D1-Acteurs métier



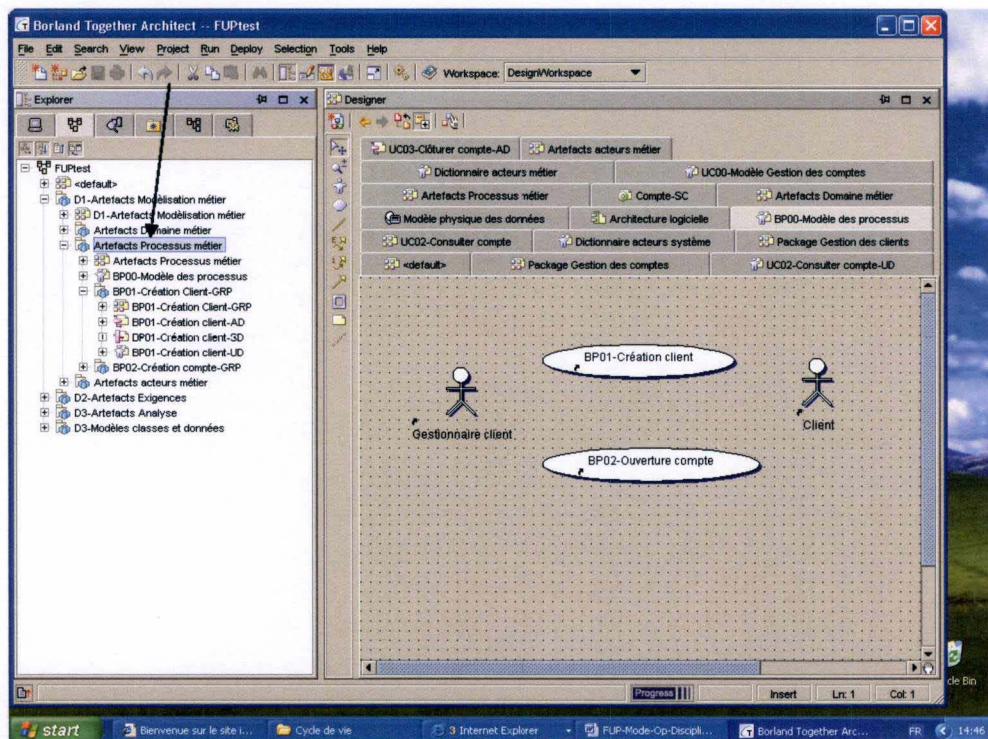
<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-Discipline Modélisation Myytier_fr v0[1].1	5/26/2006/ 0.1	8/11



### 3.4. Processus métier (O1)

Le Package 'Artefacts Processus métier' contient tous les éléments liés à la modélisation des Processus métier.

Mode opératoire détaillé : FUP-MO-D1-Processus métier

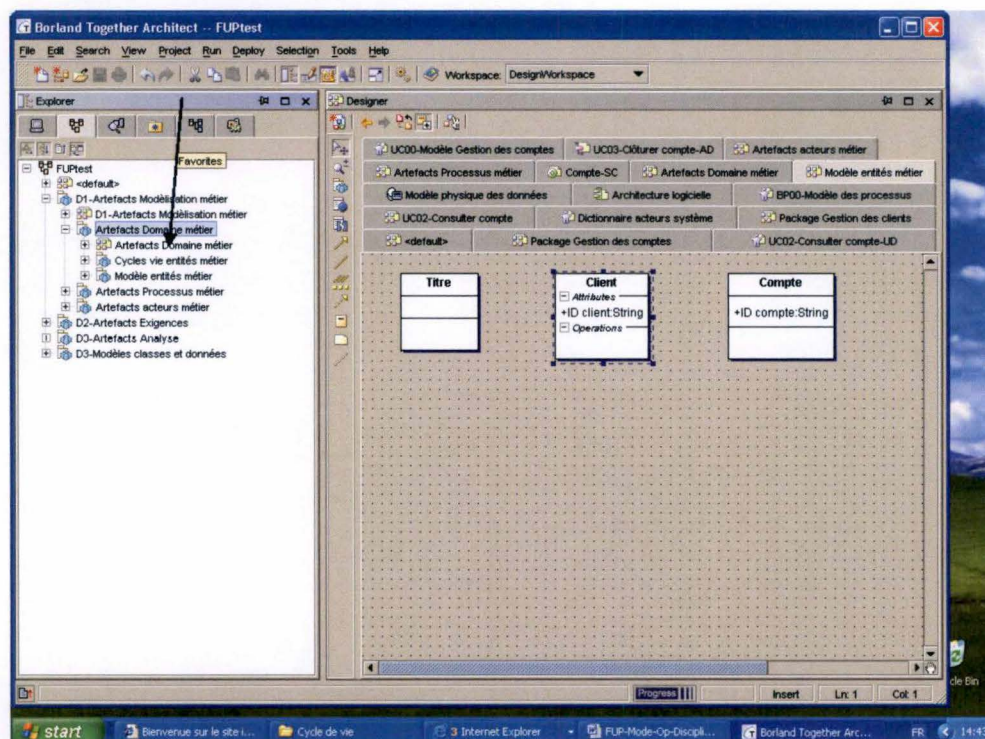


Préparé par	Nom et localisation de fichier	Date/Version	Page
	FUP-MO-Discipline Modélisation Métier_fr v0[1].1	5/26/2006/ 0.1	9/11

### 3.5. Modèle du domaine (O2)

Le Package 'Artéfacts *Domaine métier*' contient tous les éléments liés à la modélisation des entités métier.

**Mode opératoire détaillé : FUP-MO-D1-Domaine-Métier**



<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-Discipline Modélisation Myytier_fr v0[1].1	5/26/2006/ 0.1	10/11



#### 4. RECAPITULATIF DES MODES OPERATOIRES DETAILLES

FUP-MO-D1-Règles Business
FUP-MO-D1-Glossaire métier
FUP-MO-D1-Acteurs métier
FUP-MO-D1-Processus métier
FUP-MO-D1-Domaine métier

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-Discipline Modélisation Métier_fr v0[1].1	5/26/2006/ 0.1	11/11

### 12.4.3. Discipline modélisation métier – Mode opératoire détaillé – processus métier



Service Public Fédéral  
Finances

Support au développement

FUP 0.3

D1– PROCESSUS METIER

MODE OPÉRATOIRE

#### Statut

document de travail	
version interne	
soumis à acceptation	X
approuvé	



## TABLE DES MATIÈRES

<b>1. INTRODUCTION .....</b>	<b>3</b>
1.1. OBJET DU DOCUMENT .....	3
1.2. UTILISATION .....	3
1.2.1. Responsable .....	3
1.2.2. Outil de gestion .....	3
1.2.3. Restitution .....	3
1.3. FORMAT DE L'ARTEFACT .....	3
<b>2. LISTE DES ACTIONS .....</b>	<b>3</b>
<b>3. ACTIONS DÉTAILLÉES .....</b>	<b>4</b>
3.1. CHOISIR LE PROJET .....	4
3.2. CRÉER UN PACKAGE PAR PROCESSUS MÉTIER (BPXX-NOMPROCESS-GRP) .....	5
3.3. CRÉER UN USE CASE DIAGRAM PAR PROCESSUS MÉTIER (BPXX-NOMPROCESS-UD) .....	6
3.4. CRÉER UN DIAGRAMME D'ACTIVITÉ ET/OU DE SÉQUENCE .....	7
3.5. CRÉER UN 'USE CASE DIAGRAM' POUR LE MODÈLE GLOBAL DES BP .....	8
3.6. PUBLIER LES MODÈLES DANS STARTTEAM .....	9

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-D1-Processus Myytier_fr v0[1].1	5/27/2006/ 0.1	2/10

## 1. INTRODUCTION

### 1.1. Objet du document

Le but de ce document est de décrire la manière de produire l'artéfact 'Processus métier'

### 1.2. Utilisation

#### 1.2.1. Responsable

L'analyste fonctionnel

#### 1.2.2. Outil de gestion

L'identification et la modélisation des processus métiers sont gérées dans Together en local dans le Package 'Artéfacts Processus Métier'. Tout modèle introduit dans Together fera ensuite l'objet d'une publication dans StarTeam.

#### 1.2.3. Restitution

Où	Mode opératoire connexe
Dossier Analyse Métier	xxx
Liste des acteurs métier	xxx

### 1.3. Format de l'artéfact

Acteur métier/Collaborateur métier	Description
....	....

## 2. LISTE DES ACTIONS

Check-list décrivant toutes les actions nécessaires à la gestion des processus métier.

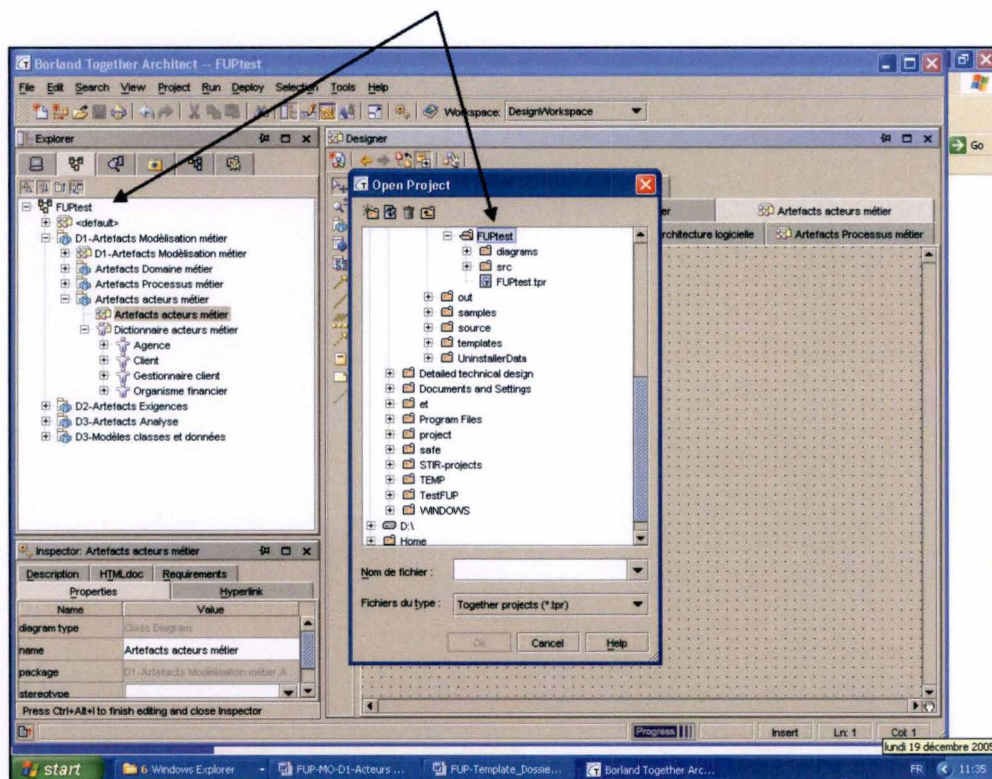
1	Choisir le projet
2	Créer un package par Business Process
3	Créer un 'use case diagram' pour modéliser le Business Process (BPxx-NomProcess-GRP)
4	Créer un 'activity diagram' par Business process
5	Créer un 'sequence diagram' par Business Process
6	Créer un 'use case diagram' pour le modèle global BP (BP00-Modèle des processus)
7	Publier les modèles dans StarTeam

Préparé par	Nom et localisation de fichier	Date/Version	Page
	FUP-MO-D1-Processus Mýýtier_fr v0[1].1	5/27/2006/ 0.1	3/10



### 3. ACTIONS DÉTAILLÉES

#### 3.1. Choisir le projet

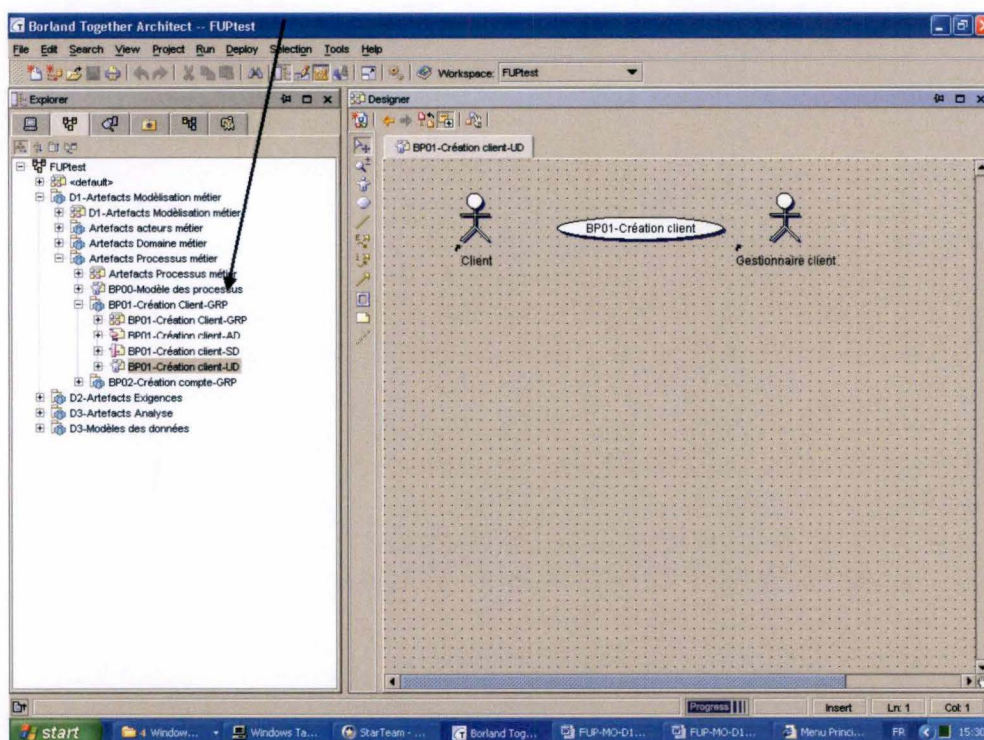


#### 3.2.

<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-D1-Processus Mýýtier_fr v0[1].1	5/27/2006/ 0.1	4/10

### **Créer un Package par processus métier (BPxx-NomProcess-GRP)**

- Créer le Package 'Artefacts processus métier' s'il n'existe pas encore
- Créer un Package BPxx-NomProcess-GRP pour regrouper tous les modèles liés un même processus métier

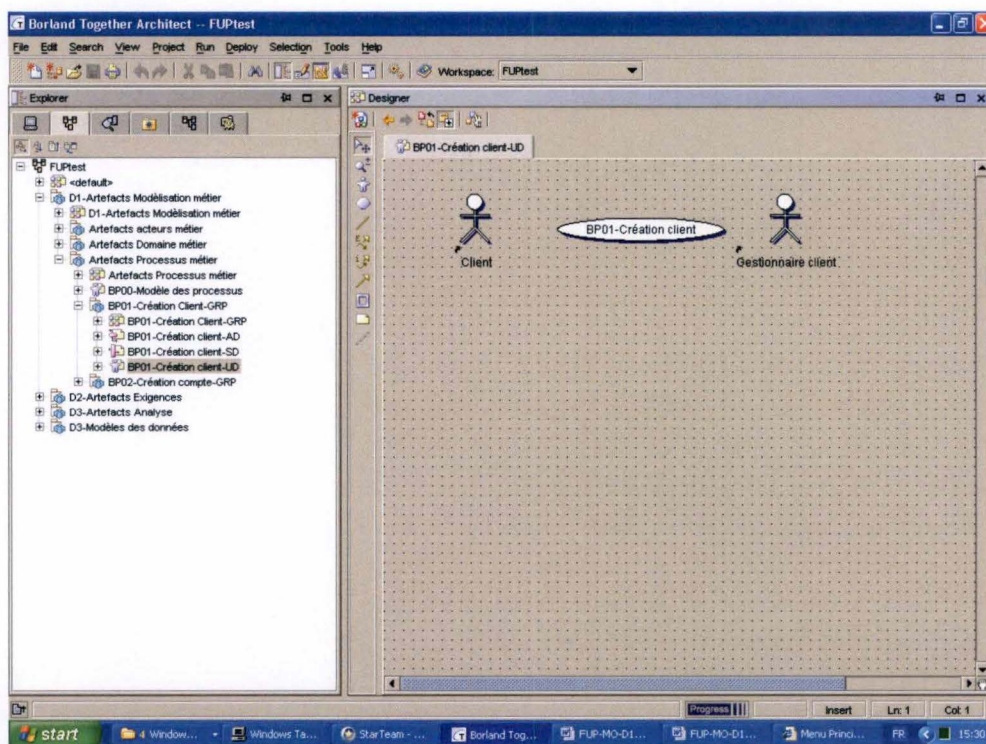


<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-D1-Processus Mýýtier_fr v0[1].1	5/27/2006/ 0.1	5/10



### 3.3. Créer un use case diagram par processus métier (BPxx-NomProcess-UD)

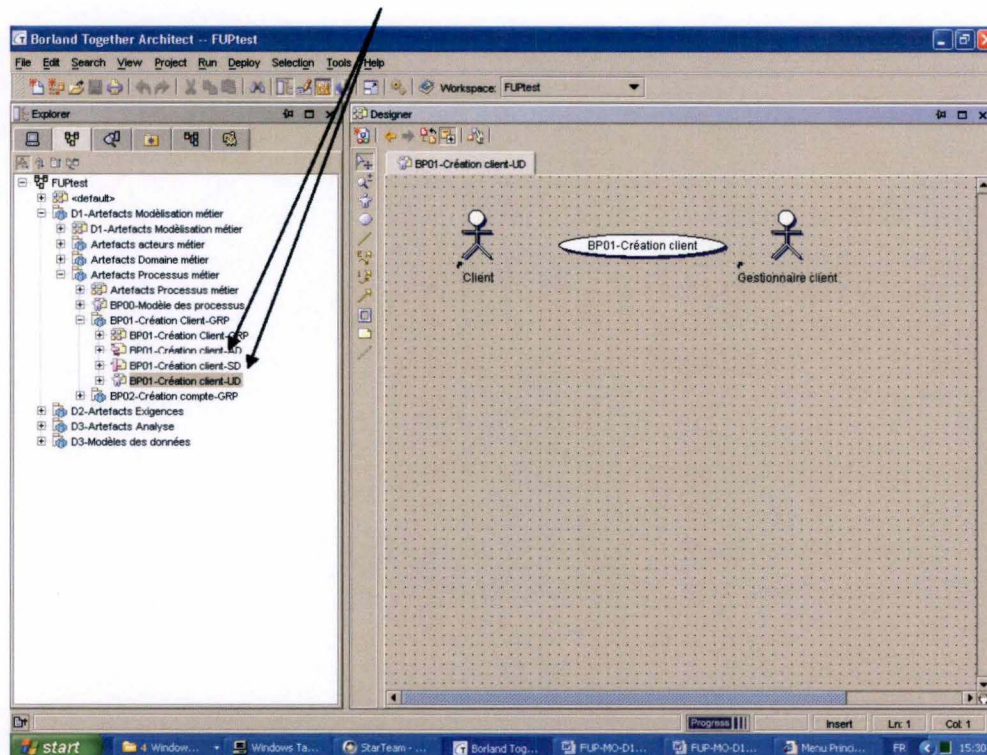
- Créer le Use case diagram modélisant les interactions processus/acteur métier
- Utiliser des shortcuts vers les acteurs métiers
- Créer le processus comme Use case BPxx-NomProcess
- détailler le processus dans l'onglet 'description'



<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-D1-Processus Mýýtier_fr v0[1].1	5/27/2006/ 0.1	6/10

### 3.4. Créer un diagramme d'activité et/ou de séquence

- a) Créer un diagramme d'activité BPxx-NomProcess-AD
- b) Créer un diagramme de séquence BPxx-NomProcess-SD



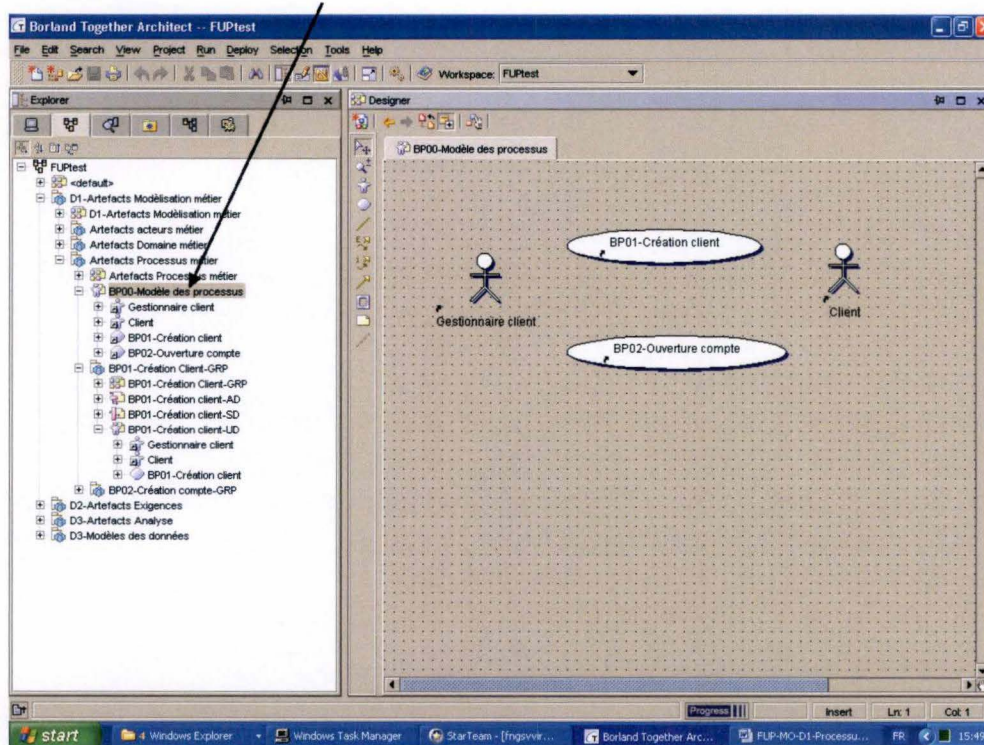
<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-D1-Processus Mýýtier_fr v0[1].1	5/27/2006/ 0.1	7/10



### 3.5. Créer un 'use case diagram' pour le modèle global des BP

Créer le modèle global en utilisant

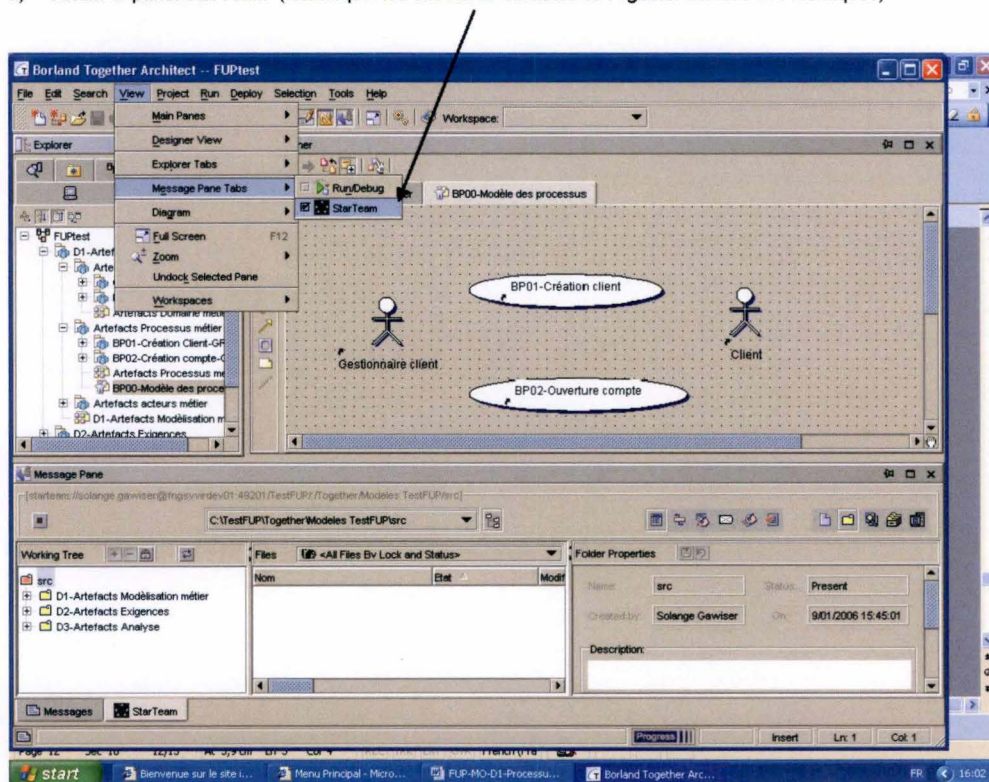
- les shortcuts vers les acteurs métier
- les shortcuts vers les processus métier (BPxx-NomProcess)



<u>Préparé par</u>	<u>Nom et localisation de fichier</u>	<u>Date/Version</u>	<u>Page</u>
	FUP-MO-D1-Processus Mýýtier_fr v0[1].1	5/27/2006/ 0.1	8/10

### 3.6. Publier les modèles dans StarTeam

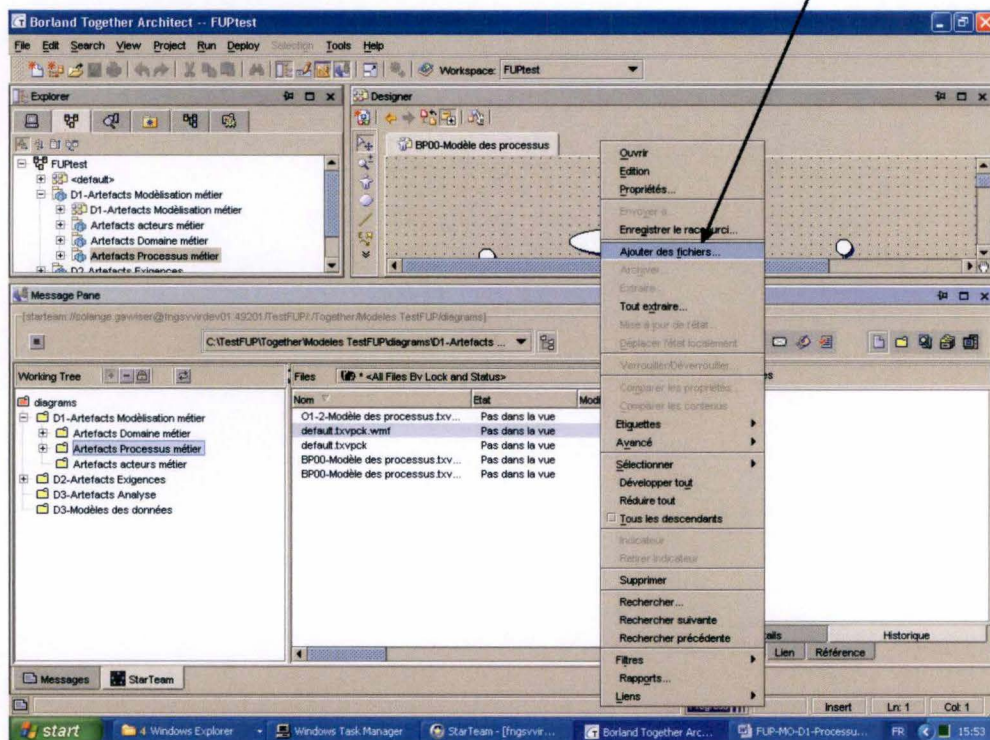
- a) Choisir le panel StarTeam (Remarque: les structures Starteam et Together doivent être identiques)



Préparé par	Nom et localisation de fichier	Date/Version	Page
	FUP-MO-D1-Processus Mýýtier_fr v0[1].1	5/27/2006/ 0.1	9/10



- b) Choisir l'action 'ajouter fichier' si le(s) fichier(s) n'existe(nt) pas encore
- c) Choisir l'action 'archiver' si le(s) fichier(s) existe(nt) déjà
- d) Effectuer les mêmes opérations pour les 'diagrammes' et les 'sources'



Préparé par	Nom et localisation de fichier	Date/Version	Page
	FUP-MO-D1-Processus Mýtétier_fr v0[1].1	5/27/2006/ 0.1	10/10